



# INTEGRATION PACK FOR IVANTI SERVICE MANAGER

*For Microsoft System Center Orchestrator*

## User Guide

Version 1.3

# Kelverion Integration Pack for Ivanti Service Manager

Copyright 2021 Kelverion Inc. All rights reserved.

Published: March 2022

[\*Feedback\*](#)

Send suggestions and comments about this document to [support@kelverion.com](mailto:support@kelverion.com)

# Contents

Installation and Configuration .....	4
System Requirements.....	4
Registering and Deploying the Integration Pack .....	4
Licensing the Integration Pack.....	5
Connecting to Ivanti .....	5
Prerequisite Configuration .....	6
Working with Configuration Files .....	6
Deploying Your Changes .....	7
Configuration File Schema .....	7
Ivanti Service Manager Activities .....	14
Common Configuration Instructions for All Activities.....	14
Activity Properties.....	14
General Tab.....	14
Properties Tab.....	14
Filters Tab.....	15
Run Behavior Tab.....	15
Published Data .....	16
Create Business Object Activity .....	17
Create Relationship Activity.....	18
Delete Business Object Activity .....	19
Delete Relationship Activity.....	20
Download Attachment Activity.....	21
Get Business Objects Activity.....	21
Get Relationship Activity.....	22
Monitor Business Object Activity.....	24
Update Business Object Activity .....	25
Upload Attachment Activity.....	25

# Installation and Configuration

---

The Integration Pack for Ivanti Service Manager is an add-on for System Center Orchestrator that enables you to integrate with Ivanti and automate service management processes.

## System Requirements

The Integration Pack for Ivanti Service Manager requires the following software to be installed and configured prior to implementing the integration.

- Microsoft System Center Orchestrator \*
- Ivanti Service Manager Cloud version
  - 2021.4.0.2021111101
- Microsoft .NET Framework 4.6.2

\* See [Kelverion.com/orchestrator](https://kelverion.com/orchestrator) for the latest Orchestrator support information.

## 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](https://technet.microsoft.com/en-us/library/hh420346.aspx) (<https://technet.microsoft.com/en-us/library/hh420346.aspx>).

### *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. Confirm that the file is not set to **Read Only**.
2. Start the **Deployment Manager**.
3. 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.
4. Click **Next**.
5. In the **Select Integration Packs or Hotfixes** dialog box, click **Add**.
6. Locate the **.OIP** file that you copied locally from step 1, click **Open** and then click **Next**.
7. In the **Completing the Integration Pack Wizard** dialog box, click **Finish**.
8. On the **End User Agreement** dialog box, read the Keverion License Terms, and then click **Accept**.
9. 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 Keverion license before running any runbooks that contain activities from the integration pack

#### *To deploy the integration pack license file:*

1. Copy the **.KAL** license file to %PROGRAMFILES(X86)%\Keverion Automation\Licenses
2. Repeat for each Orchestrator Runbook Server and Runbook Designer host system.

## Connecting to Ivanti

The Keverion Integration Pack for Ivanti Service Manager requires an Ivanti API Key to authenticate with the Ivanti REST Web Service. Following best practices, it is advisable to create a new API Key specifically for use with the Integration Pack.

#### *To create a new API key:*

1. Login in to Ivanti Service Manager,
2. Open the Ivanti Settings by clicking the **wrench** icon in the top right corner.
3. Expand **Security Controls** in the **Configure** section.
4. Select **API Keys**.

5. Click **Add Key Group**.
6. Enter an appropriate **Name** and **Description**.
7. Click **Save Key Group**.
8. Click **Back**.
9. Select the desired key group and click **Add API Key**.
10. Select a **User** and **Role** that is appropriate for the work that your runbooks will perform.
11. Click **Save Key**.
12. Click **Back**.
13. Copy the new API Key so that you can configure System Center Orchestrator.

## Prerequisite Configuration

Prerequisite Configurations establish reusable links between Orchestrator and a specific Ivanti Service Manager instance. You can create as many configurations as you require specifying links multiple Ivanti servers. You can also create multiple configurations to the same server to allow for different users and/or roles.

### *To set up a prerequisite configuration in Runbook Designer*

1. On the **Options** menu, click **KA Ivanti Service Manager**.
2. In the **KA Ivanti Service Manager** dialog box, click the **Configuration** tab and then click **Add**.
3. On the **Configuration** tab, click **Add**.
4. In the **Add Configuration dialog** box, click the browser next to the **Type** box. Select **Ivanti Service Manager Configuration** and then click **OK**.
5. In the **Configuration File Path** box, type the file path to the desired configuration file. A default configuration file is provided for convenience.
6. In the **Server URL** box, type the URL of the Ivanti Service Manager instance. For example, `https://myserver.vantosi.com`
7. In the **REST API Key** box, type the Ivanti REST API Key. See [Connecting to Ivanti](#) for more information on API Keys.
8. Click **OK**, and then click **Finish**.

## Working with Configuration Files

Integration with Ivanti Service Manager is supported by a JSON (JavaScript Object Notation) formatted configuration file. The configuration file contains business object definitions for the supported Ivanti Service Manager business objects. This information is used to support discovery in System Center Orchestrator and to execute runbooks at runtime. The integration pack is deployed with a default configuration file, supporting some important business object types, for your convenience.

The configuration file can also be used to customize the integration with your Ivanti Service Manager instance. Customizations can include adding new field definitions to existing Ivanti Service Manager business objects, such as Event or Change, or adding definitions for new business objects that are not supported by default.

## Deploying Your Changes

The **Configuration File Path** property from the integration pack configuration options specifies the location of the JSON configuration file that the integration pack will be using. By default, this specifies the configuration file that is installed with the integration pack. If you wish to make customizations to the configuration file, we recommend making a copy of the default configuration file, saving it in a safe location, and then updating the **Configuration File Path** property accordingly.

**Important:** When you deploy an updated version of the Integration Pack, the default configuration file will be overwritten. Therefore, it is important that you copy your custom files to a safe location that is unaffected by the update.

## Configuration File Schema

The business object definitions in the configuration file used by the Integration Pack for Ivanti Service Manager are formatted using JSON. The following JSON Schema outlines the JSON used for configuration files.

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://kelverion.com/ivanti-configuration.schema.json",
  "title": "Integration pack for Ivanti Service Manager Configuration File",
  "type": "object",
  "properties": {
    "Lists": {
      "description": "Collection of approved values for Ivanti Service Manager fields",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Name": {
            "description": "The name of an Ivanti Service Manager field",
            "type": "string"
          },
          "Values": {
            "description": "The collection of approved values for the field",
            "type": "array",
            "items": {
              "type": "string"
            }
          }
        }
      }
    },
    "required": ["Name", "Values"]
  },
}
```

```

    "uniqueItems": "true"
  },
  "BusinessObjects": {
    "description": "Collection of Ivanti Business Object definitions",
    "type": "object",
    "items": {
      "type": "object",
      "properties": {
        "DisplayName": {
          "description": "The name used to display the Business Object type",
          "type": "string"
        },
        "Name": {
          "description": "The name used in the Ivanti REST API",
          "type": "string"
        },
        "NamePlural": {
          "description": "The name used for collections in the Ivanti REST API",
          "type": "string"
        },
        "BaseBusinessObject": {
          "description": "The name of the business object that this type extends",
          "type": "string"
        },
        "IsAbstract": {
          "description": "Indicates whether objects of this type can be created",
          "type": "boolean"
        },
        "CanCreate": {
          "description": "Indicates whether objects of this type can be created",
          "type": "boolean"
        },
        "CanUpdate": {
          "description": "Indicates whether objects of this type can be modified",
          "type": "boolean"
        },
        "CanGet": {
          "description": "Indicates whether objects of this type can be retrieved",
          "type": "boolean"
        },
        "CanDelete": {

```



```

    "description": "Indicates whether objects of this type can be deleted",
    "type": "boolean"
  },
  "CanMonitor": {
    "description": "Indicates whether objects of this type can be monitored",
    "type": "boolean"
  },
  "CreateTolerance": {
    "description": "Time in seconds to differentiate new and modified objects",
    "type": "integer"
  },
  "Fields": {
    "description": "Collection of field definitions for the business object",
    "type": "array",
    "items": {
      "type": "object",
      "properties": {
        "DisplayName": {
          "description": "The name used to display the field in Orchestrator",
          "type": "string"
        },
        "Name": {
          "description": "The name used to identify the field in Ivanti",
          "type": "string"
        },
        "DataType": {
          "description": "The data type used for the field",
          "type": "string"
        },
        "Required": {
          "description": "Is the field required for new objects"
        },
        "ReadOnly": {
          "description": "is the field read only",
          "type": "boolean"
        },
        "IsFilter": {
          "description": "Can the field can be used for filtering",
          "type": "boolean"
        },
        "IsId": {

```

```

        "description": "is the field a unique identifier",
        "type": "boolean"
    },
    "PublishedOnCreate": {
        "description": "Is the field published for new objects",
        "type": "boolean"
    },
    "ValueListName": {
        "description": "The name of the list used for field values",
        "type": "string"
    }
},
"required": ["DisplayName", "Name", "DataType" ]
}
},
"Relationships": {
    "description": "Collection of relations supported by the business object",
    "type": "array",
    "items": {
        "type": "object",
        "properties": {
            "DisplayName": {
                "description": "The name used to display the relationship",
                "type": "string"
            },
            "Name": {
                "description": "The name used to identify the relationship",
                "type": "string"
            },
            "RelatedBusinessObjectIdDisplayName": {
                "description": "The property used to identify the child object",
                "type": "string"
            },
            "RelatedBusinessObject": {
                "description": "The name of the related business object type",
                "type": "string"
            },
            "CanCreate": {
                "description": "Can you create this relationship in Orchestrator",
                "type": "boolean"
            }
        }
    }
},

```

```

        "CanGet": {
            "description": "Can you retrieve this relationship in Orchestrator",
            "type": "boolean"
        },
        "CanDelete": {
            "description": "Can you delete this relationship in Orchestrator",
            "type": "boolean"
        },
    },
    "required": [
        "Name",
        "DisplayName",
        "RelatedBusinessObjectIdDisplayName",
        "RelatedBusinessObject"
    ]
},
"uniqueItems": true
}
},
"required": ["Name", "NamePlural", "DisplayName", "Fields"]
},
"uniqueItems": "true"
}
}
}

```

## Defining Business Objects

Business object definitions provide the integration pack with the information used to integrate with Ivanti Service Manager.

### Name Property

The **Name** property specifies the internal name of the business object that is being defined. This value must match the business object name defined in Ivanti Service Manager under **Settings/Build/Business Objects**. For example, *Incident* or *Change*.

### NamePlural Property

The **Name** property specifies the plural name for the business object. For example, *Incidents* or *Changes*.

### DisplayName Property

The **DisplayName** property specifies the name used to display the business object in System Center Orchestrator.

### BaseBusinessObject Property

The **BaseBusinessObject** property specifies the business object in the configuration file that this business object extends. The definition of the base business object must precede the definition of the inherited business object, in the configuration file.

### Fields Property

The **Fields** property is used to define the fields that are associated with the business object. See the [Defining Business Object Fields](#) section for more information on defining fields.

### Relationships Property

The **Relationships** property is used to define the business object relationships that are supported by the integration pack. See the [Defining Business Object Relationships](#) section for more information on defining relationships.

### *Defining Business Object Fields*

Field definitions provide the integration pack with information used to support specific business object fields. This information is be used by integration pack activities to present appropriate input properties and filter options for the supported business objects.

### Name Property

The **Name** property specifies the internal name of the business object field. This value must match the field name as defined in Ivanti Service Manager under **Settings/Build/Business Objects**.

### DisplayName Property

The **DisplayName** property specifies the name used to display the field in System Center Orchestrator.

### DataType Property

The **DataType** property specifies the field's underlying data type. The integration pack supports the following data types:

- Boolean
- Byte
- Date
- DateTime
- DateTimeOffset
- Decimal
- Double
- Guid
- Int16
- Int32
- Int64
- String

### Required Property

The **Required** property indicates whether the field is mandatory for new objects. When set to *True* the field will become a required property when configuring the **Create Business Object** activity in System Center Orchestrator.

### ReadOnly Property

The **ReadOnly** property indicates whether the field is read only available as a published data item.

#### IsFilter Property

The **IsFilter** property indicates whether the field can be used for filtering in **Get Business Object** and **Monitor Business Object** activities.

#### IsId Property

The **IsId** property indicates whether the field can be used uniquely identify business object instances.

#### PublishOnCreate Property

The **PublishOnCreate** property indicates whether the field will be published by the **Create Business Object** activity.

#### ValueListName Property

The **ValueListName** property specifies the name of a list that is used to provide browser values for the field in System Center Orchestrator. The list must be defined in the Lists section of the configuration file.

### *Defining Business Object Relationships*

Relationship definitions provide the integration pack with information used to support managing relationships between business objects. This information is used by the relationship activities to present the appropriate input properties.

#### Name Property

The **Name** property specifies the internal name of the relationship. This value must match the relationship name as defined in Ivanti Service Manager under **Settings/Build/Business Objects**.

#### DisplayName Property

The **DisplayName** property specifies the name used to display the relationship in System Center Orchestrator.

#### RelatedBusinessObject Property

The **RelatedBusinessObject** property specifies the name of the parent business object in the relationship. This value must match the related business object name as defined in Ivanti Service Manager under **Settings/Build/Business Objects**.

#### RelatedBusinessObjectIdDisplayName Property

The **RelatedBusinessObjectIdDisplayName** property specifies the display name, used for properties in Orchestrator, to identify the child object in the relationship.

# Ivanti Service Manager Activities

---

This integration pack adds the KA Ivanti Service Manager category to the Runbook Designer Activities pane. This category contains the following activities:

- Create Business Object
- Create Relationship
- Delete Business Object
- Delete Relationship
- Download Attachment
- Get Business Object
- Get Relationship
- Monitor Business Object
- Update Business Object
- Upload Attachment

## Common Configuration Instructions for All Activities

The following configuration instructions apply to all activities in this integration pack.

### 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 Runbook Designer.

#### *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, **Properties** tab, and a **Run Behavior** tab. Some activities may have additional tabs, such as a **Filters** tab.

### 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 Tab

The Properties 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 which prerequisite configuration to use to connect to Ivanti Service Manager.

#### *To configure the Configuration Name property:*

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

## Filters Tab

The Monitor Business Object and Get Business Object activities include a filters tab. You can use the Filters tab to add one or more filters to target a specific subset of business objects. Each filter includes **Name**, **Relation**, and **Value** properties, that you must configure.

#### *To add a filter to your runbook activity:*

1. Click the **Filters** tab and then click **Add**.
2. In the **Name** box, select the business object field that the filter will target.
3. In the **Relation** box, select the relation used to evaluate the filter value.
4. In the **Value** box, type the value that you want the filter to evaluate.
5. Click **OK**.

The relations that are available when configuring a filter are dependent on the data type of the business object field that the filter is targeting. One or more of the following will be provided.

- **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.

## Run Behavior Tab

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

### *Published Data Behavior*

By default, Published Data is passed as multiple individual outputs. You can alternatively specify that all values be flattened into a single comm-delimited value (.csv) file.

When you enable the Flatten feature, you also choose a multi-value 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.

**Note:** The Flatten feature does not flatten data across multiple instances of the same activity. It only flattens multiple values returned from a single instance of the 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#EventNotifications) (https://technet.microsoft.com/en-us/library/hh489610.aspx#Event Notifications).

### *Published Data*

Published 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 subscribe only 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.

#### *To subscribe to Published Data of an earlier activity in the workflow:*

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. By default, the dialog box only displays Published Data that is specific to that activity. To include Published Data that is common to all activities, click **Show Common Published Data**.
3. Click the published data item 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 see [Published Data](http://technet.microsoft.com/en-us/library/hh403821.aspx) (http://technet.microsoft.com/en-us/library/hh403821.aspx).



# Create Business Object Activity

---

The **Create Business Object** activity is used to create and initialize a new business object.

## *Required Properties*

You must select the type of business object that you want to create. After you have selected business object type, the activity will provide additional properties, that you must configure, to initialize the new object.

---

<b>Business Object Type</b>	The type of business object to create.
-----------------------------	--

---

## *Optional Properties*

The activity will provide optional properties that correspond to the non-mandatory fields in the business object type that you selected, and these can be configured as needed to initialize the new object.

## *Published Data*

The activity publishes the following activity specific data items.

---

<b>&lt;Business Object Type&gt; ID</b>	The unique ID of the business object that was created.
<b>&lt;Business Object Type&gt; Number</b>	The display number of the business object that was created. Included for Incident, Change and Event objects.
<b>Asset ID</b>	The Asset ID of the configuration item (CI) that was created. Included for Computer and Enterprise Application objects.

---

# Create Relationship Activity

---

The **Create Relationship** activity is used to create a relationship between two business objects.

## *Required Properties*

You must configure the following properties.

<b>Business Object Type</b>	The parent business object type.
<b>&lt;Business Object Type&gt; ID</b>	The unique ID of the parent business object
<b>Relationship</b>	The type of relationship to create.
<b>&lt;Related Business Object Type&gt; ID</b>	The unique ID of the child business object.

## *Published Data*

The activity publishes the following activity specific data items.

<b>&lt;Business Object Type&gt; ID</b>	The unique ID of the parent business object.
<b>&lt;Related Business Object Type&gt; ID</b>	The unique ID of the child business object.

# Delete Business Object Activity

---

The **Delete Business Object** activity is used to delete a business object.

## *Required Properties*

You must configure the following properties.

<b>Business Object Type</b>	The type of business object to delete.
<b>&lt;Business Object Type&gt; ID</b>	The unique ID of the business object to delete.

## *Published Data*

The activity publishes the following activity specific data items.

<b>&lt;Business Object Type&gt; ID</b>	The unique ID of the business object that was deleted.
<b>&lt;Business Object Type&gt; Number</b>	The display number of the business object that was deleted. Included for deleting Incident, Change and Event objects.

# Delete Relationship Activity

---

The **Delete Relationship** activity is used to delete a relationship between two business objects.

**Important:** Your Ivanti Service Manager instance may have business rules in place that prevent the deletion of some business object relationships.

**Important:** Deleting a relationship between a parent and child object, may cause the child object to become orphaned. For example, deleting the Service Request Contains Journal relationship between a Service Request and Journal Note will orphan the Journal Note because it can no longer be accessed directly without knowing its specific Journal ID.

## Required Properties

You must configure the following properties.

<b>Business Object Type</b>	The parent business object type.
<b>&lt;Business Object Type&gt; ID</b>	The unique ID of the parent business object.
<b>Relationship</b>	The type of relationship to delete.
<b>&lt;Related Business Object Type&gt; ID</b>	The unique ID of the child business object.

## Published Data

The activity publishes the following activity specific data items.

<b>&lt;Business Object Type&gt; ID</b>	The unique ID of the parent business object.
<b>&lt;Related Business Object Type&gt; ID</b>	The unique ID of the child business object.

## Download Attachment Activity

The **Download Attachment** activity is used to download an attachment.

**Important:** You can use the **Get Business Object** activity to retrieve existing **Attachment** records, and you can filter by **Parent ID** to retrieve the attachments associated with a business object.

### Required Properties

You must configure the following properties.

<b>Attachment ID</b>	The unique ID of the attachment that is to be downloaded.
<b>Download Folder</b>	The folder where the attachment file will be downloaded.

### Optional Properties

The activity publishes the following optional properties.

<b>If Attachment Exists</b>	<p>Specifies how the attachment file should be saved, in the case when a file with the same name already exists in the Download Folder. Available options are:</p> <ul style="list-style-type: none"><li>• <b>Overwrite</b> – the existing file will be overwritten with the attachment file.</li><li>• <b>Rename</b> – the attachment file will be saved with a new name. For example, if Attachment.doc exists, the new attachment file name will be Attachment (1).doc.</li><li>• <b>Ignore</b> – The attachment will not be saved. The published value for the File Path output property will be null.</li></ul> <p>By default, the activity overwrites existing files, if this property is not specified.</p>
-----------------------------	--

### Published Data

The activity publishes the following activity specific data items.

<b>Attachment ID</b>	The unique ID of the attachment that was downloaded.
<b>File Path</b>	The file path of the downloaded attachment.

## Get Business Objects Activity

The **Get Business Objects** activity is used to retrieve business object information. Filters can be combined to select a specific collection of business objects.

### Required Properties

You must configure the following properties.

<b>Business Object Type</b>	The type of business object to retrieve.
<b>Limit</b>	The maximum number of business objects to retrieve. The maximum value is one hundred.

### *Optional Properties*

You can use the following properties, as necessary, to control the behavior of the activity.

<b>Descending</b>	Indicates that the published objects should be in descending order. By default, results are published in ascending order.
<b>Order By</b>	Specifies the field that should be used to order the published objects.
<b>Skip</b>	Specifies the number of records to skip.

### *Filters*

The activity provides filters that correspond to the fields in the business object type that you selected. You can combine one or more filters to selectively control which business objects to retrieve.

### *Published Data*

The activity publishes data that represents the business object records that were retrieved. Each record has published data items that correspond to the fields in business object that you selected. The activity also publishes the following activity specific data items.

<b>Object Count</b>	The number of business object records that were retrieved and published. A maximum of one hundred records can be retrieved from Ivanti Service Manager.
<b>Total Object Count</b>	The total number of business objects matching the specified filters.

**Important:** A **Total Object Count** greater than one hundred, indicates that the **Get Business Object** activity was only able to retrieve and publish a subset of the larger set of business objects that matched your filter criteria.

## Get Relationship Activity

The **Get Relationship** activity is used to retrieve information about a business object's relationships.

### *Required Properties*

You must configure the following properties.

<b>Business Object Type</b>	The parent business object type.
<b>&lt;Business Object Type&gt; ID</b>	The unique ID of the parent business object.
<b>Relationship</b>	The name of the relationships to retrieve.

### *Published Data*

The activity also publishes the following activity specific data items.

<b>&lt;Business Object Type&gt; ID</b>	The unique ID of the parent business object.
<b>&lt;Related Business Object Type&gt; ID</b>	The unique ID of the child business object.
<b>Relationship Count</b>	The number of relationships that were retrieved.

# Monitor Business Object Activity

The **Monitor Business Object** activity is used to monitor for new and/or updated business objects.

## *Required Properties*

You must configure the following properties.

<b>Business Object Type</b>	The type of business object to monitor
<b>Limit</b>	Specifies that maximum number of records to retrieve. The maximum value is one hundred.
<b>Monitor New</b>	Indicates whether the monitor triggers <i>on</i> new business object records.
<b>Monitor Updated</b>	Indicates whether the monitor triggers on updated business object records.
<b>Monitor Interval (seconds)</b>	The monitor polling interval in seconds.
<b>Create Tolerance (seconds)</b>	The interval in seconds, following the creation of a business object, during which subsequent updates will be ignored.

## *Filters*

The activity provides filters that correspond to the fields in the business object type that you selected. You can combine one or more filters to selectively control which records to retrieve. If no filters are selected, the activity will return all records in the table, up to the specified limit.

## *Published Data*

The activity publishes data that represents the records that were retrieved, and each record has published data items that correspond to the fields in Ivanti business object that you selected. The activity also publishes the following data items.

<b>Object Count</b>	The number of business object records that were published.
<b>Total Object Count</b>	The total number of business objects matching the specified filters.

**Important:** A **Total Object Count** greater than one hundred, indicates that the monitor is publishing a subset of the business object records that triggered the monitor, based on the time that they were created or modified and any filters that you configured. This is because the monitor can only retrieve a maximum of one hundred records each time it polls Ivanti Service Manager for changes.



## Update Business Object Activity

---

The **Update Business Object** activity is used to update one or more fields in an existing business object.

### *Required Properties*

You must configure the following properties.

<b>Business Object Type</b>	The type of business object to update.
<b>&lt;Business Object Type&gt; ID</b>	The unique ID of the business object to update.

### *Optional Properties*

The activity will provide parameters that correspond to the fields in the business object type that you selected, and these can be used, as necessary, to update the specified business object.

### *Published Data*

The activity publishes the following activity specific data items.

<b>&lt;Business Object Type&gt; ID</b>	The unique ID of the business object that was updated.
--	--

## Upload Attachment Activity

---

The **Upload Attachment** activity is used to upload an attachment and associate it with an existing business object.

**Important:** A new attachment record will be created and linked to the specified business object.

### *Required Properties*

You must configure the following properties.

<b>Business Object Type</b>	The type of business object the attachment will be associated with.
<b>&lt;Business Object Type&gt; ID</b>	The unique ID of the business object the attachment will be associated with.
<b>File Path</b>	Specifies the location of the attachment file that will be uploaded.

### *Published Data*

The activity publishes the following activity specific data items.

<b>Attachment ID</b>	The unique ID of new attachment record.
<b>&lt;Business Object Type&gt; ID</b>	The unique ID of the business object the attachment was associated with.