STIBO SYSTEMS
MASTER DATA MANAGEMENT

USER GUIDE

Workflows

Rel 10.2-MP3 (September 21, 2021)

Table of Contents

Workflows

This guide concerns designing, using, and monitoring STEP Workflows in the STEP Workbench. Workflows are managed within the workbench, though end users can access them via both workbench and Web UI. For information on configuring and working with workflows in Web UI, see the **Workflows in Web UI** section of the **Web User Interfaces** documentation.

Workflows may be considered as a view or representation of real work within a company, allowing businesses to manage and streamline the flow of tasks among varying groups, users, and system activities. A workflow definition consists of a number of states that an object or task can be in at a given time, the legal transitions between these states, and the events via which the transitions can be triggered. The flow being described may refer to an asset, service, person, product, or other element that is being transferred from one state to another. For example, introduction of a new product may involve work from a variety of business groups, such as Product Management, Marketing, Creative, and Legal. Each group needs to interact with the product in some way, and likely have differing sets of data required for this interaction. Some groups may not be able to do their work until another group has completed theirs, while other groups may be able to act on the product in parallel.

The same principles apply for any other data types and any type of data maintenance. Regardless of the type of data or process being handled, an object in a state represents a task that requires work. For instance, a user could be required to populate a set of attributes, and when completed, manually trigger a transition from one state to the next. Alternatively, an object could be in a state waiting for data from an external system, waiting for a background process to complete, or waiting for a specific event elsewhere in the workflow or in STEP in general, and thus not require any direct human interaction.

STEP Workflows are data-centric workflows, meaning that a workflow instance is defined as the relation between a workflow definition and an object in the STEP database. An object can be in multiple workflows simultaneously, and each instance will have no impact on the others (unless the workflow definitions include business rules to create a relationship between various workflows).

On top of the basic STEP Workflow functionality, a number of additional features are available that make workflows powerful tools for streamlining processes and ensuring data quality. These include, but are not limited to, options to:

- Configure business rules to validate data, assign tasks, move tasks through workflows, integrate workflows with other processes such as imports and exports, and more
- Configure data views specific for each workflow state, allowing each actor in the flow to focus only on the data most relevant for their tasks
- Support both static and dynamic task assignment and ownership, ensuring that the right actor sees the right task at the right time
- Design for sequential and parallel execution of tasks, streamlining work processes for maximum efficiency
- Set deadlines and escalation actions, ensuring that tasks are completed in a timely manner
- Configure dashboards for snapshot monitoring and KPI reporting, allowing administrators to quickly view and assess key data points across the entire workflow

The supported workflow enabled object types are:

- Assets
- Attributes
- Classifications
- Entities
- Lists of Values (LOVs)
- Planned pages
- Products
- Publications
- Publication sections

While workflows can become quite complex, the basics are really simple, and it is possible to get a small functional workflow up and running in a matter of minutes. This guide will begin with the basic material and work up to more advanced topics.

Getting Started with STEP Workflows

This portion of the material introduces the user to basic STEP Workflow functionality, including creating a simple workflow and accessing tasks within it. More advanced material can be found in the **Advanced Workflow Topics** section of the **Workflows** documentation, but it is recommended that users be well-versed in the introductory topics before beginning advanced functions.

If you are new to workflows, it is recommended that you read the material in the suggested order, as topics build upon one another.

However, if you are concerned only with the end user functionality (working with tasks in a workflow), as opposed to designing and configuring a workflow, you may wish to skip directly to the **Working with Tasks in Workflows** section of the material.

Suggested Reading Order:

1. Workflow Terminology
2. Workflow Privileges
3. Initial Setup for Workflows
4. Creating a Workflow
5. STEP Workflow Designer
6. Views and Mappings for Workflows
7. Working with Tasks in Workflows
8. Managing Workflows

Workflow Terminology

This defines a list of commonly used terms in regards to workflows.

Action	Type of business rule that can be executed on arrival in a new state (on-entry), on exit from a state (on-exit), or on the transition between two states (on-transition). Actions may modify data.
Assignee	The user or group to which a task is assigned for work to be performed. User states can be assigned directly to a named user or to a user group. When a state is assigned to a user group the task is 'offered' to all members of that group. To perform the task associated with the state, a member of the group has to claim the task. Once a task is claimed it is no longer visible in the group task list; only the person who claimed the task is able to see it. A user may subsequently release a task back to the group.
Automated State	State in which the STEP system performs the required work and automatically progresses the workflow when the automated work is complete. Typically requires the use of at least one business rule.
Business Rule	Automated activity that may implement customer-defined rules, such as validations, automatic updates, notifications, data-driven workflow routing, etc. Also see Condition and Action.
Condition	Type of business rule that can prevent a transition from being taken when an event is attempted. Within a workflow, conditions may only be executed on attempted transitions (e.g., before 'Proceed' event is allowed, validate that all required data has been populated). A condition may not change data.
Deadline	Optional setting for time allowed for a task to be completed. If a deadline expires on a task it is possible to perform an escalation.
Escalation	The action(s) to be taken when a deadline expires. Escalations require at least one business rule to be configured, which is triggered by the escalation.
Event	The name associated with a transition between two states, such as Proceed, Reject, or Accept. When a task is completed an event is triggered to transition an object to the next state(s) in the workflow. In most cases where there is just a single event, this is called 'Proceed' but in some cases there is a need to support different routing via different events (e.g., 'Accept' or 'Reject') or a different name is chosen to add clarity (e.g., 'Approve').
Final State	The state within a workflow or a parallel cluster that is the last step of this workflow or parallel cluster. A final state cannot have any outbound events as there are no available transitions from the state. Note that for parallel clusters, final states are used to auto-progress the workflow when all concurrent parallel clusters have reached their final state.

Initial State	The first state that is activated within a workflow or within a parallel cluster of a workflow.
Task	The work and/or action that must be completed within a specific workflow state (whether by a user or via automation).
Task Fields	Fields (attribute, reference, workflow variable) that are available within a specific state for the system or a user to access (view and/or populate). Task fields may or may not be included in the view.
Parallel cluster	A set of states that form a sequential branch of dependent states which may run parallel to other independent branches within the same workflow.
State	A step within a workflow to which a specific user, user group, or system can be assigned to complete the task(s) associated with the state. Also see User State and Automated State.
Transition	The change of tasks from one state to another.
User State	State that requires a user to complete a task before marking the state complete. May include some automation that happens on entry or exit of the state, but requires a user to trigger the exit from the state (indicating completion of the associated task).
View	The set of specific data points that are presented to the user in the STEP Workbench when a task is selected. Views are configurable by state and are only applicable for user states that will be accessed directly from the STEP Workbench. Views are not applicable for automated states, and states accessed via a Web UI have interfaces defined via the Web UI configuration. Within the STEP Workbench, users can still choose to use the full set of workbench data and tabs to work on workflow tasks if desired (and permitted via privilege configurations). However, a view can streamline work processes so that only a small subset of fields that are directly relevant to the task / state are displayed.
Workflow	Business process(es) modeled as a series of states and tasks that can be executed by users and/or automatic processes within the STEP system. Workflows prompt the next step of a process to be started, by either creating a task for a user, or sending an alert. Workflows may contain a number of process steps or tasks, as well as business logic.
Workflow Attachment	A temporary asset attached to a task, that does not exist outside of the workflow.
Workflow Variable	A temporary data point (similar to an attribute value) attached to a task, that does not exist outside of the workflow.

Workflow Privileges

There are a number of Setup Actions (privileges) directly related to STEP Workflows, both in terms of allowing users to configure workflows and to work with tasks within workflows. In addition to workflow-specific privileges, workflows also take into account general privileges. For example, if a user does not have access to a specific set of objects or data points, they will not have access to them within a workflow either, unless specific access is granted within the workflow.

For complete information on privileges related specifically to workflows, see the **Workflows** section of the **Setup Actions** topic in the **System Setup / Super User Guide** documentation.

For information on editing general privileges to provide access to data or objects specifically within a particular workflow state, see the **Editing Privilege Rules** section of the **System Setup / Super User Guide** documentation.

Initial Setup for Workflows

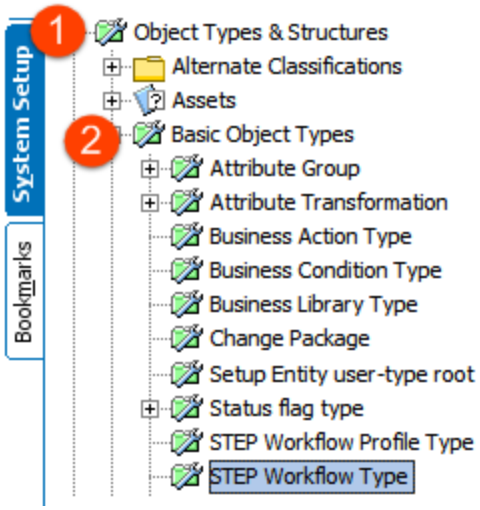
In order to create workflows in STEP, some basic configuration must first be completed. Specifically, the STEP Workflow Type basic object type must have a Setup Group parent assigned, and a System Setup object of the proper type must have been created to hold the workflows.

In most cases, this setup will already be complete on a system and is only required if a Workflows folder (or other setup node enabled to workflow children) is **not** present on the System Setup tab, as shown in step 7 below.

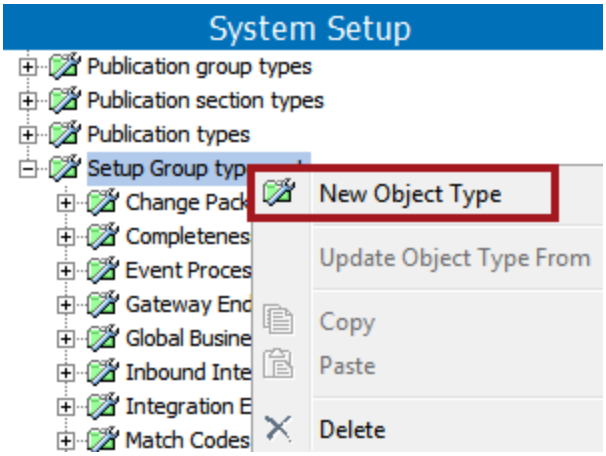
The setup described below is only required to be completed once on any system, but may be carried out multiple times if it is desired to separate workflows for organizational purposes.

Enabling a Setup Group for Workflows

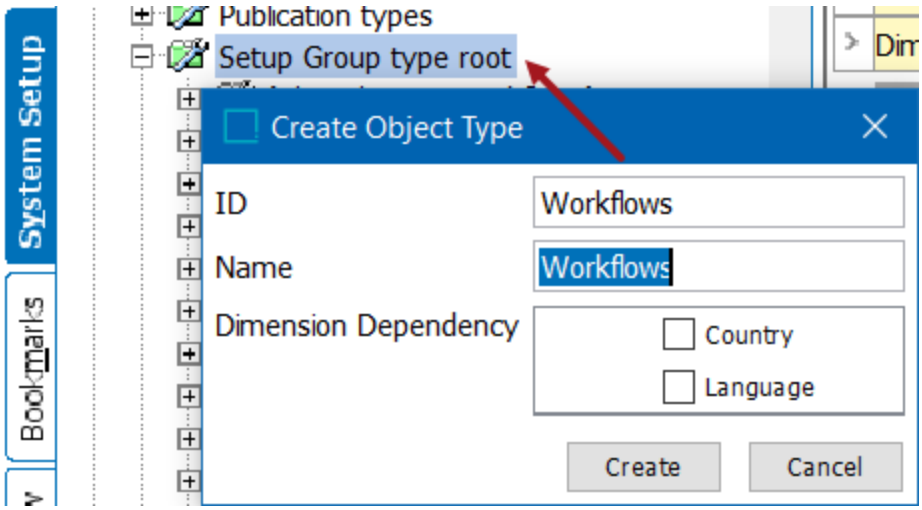
1. Go to System Setup and select **Object Types and Structures** > **Basic Object Types** and verify that **STEP Workflow Type** exists. If not, contact your system administrator for assistance.



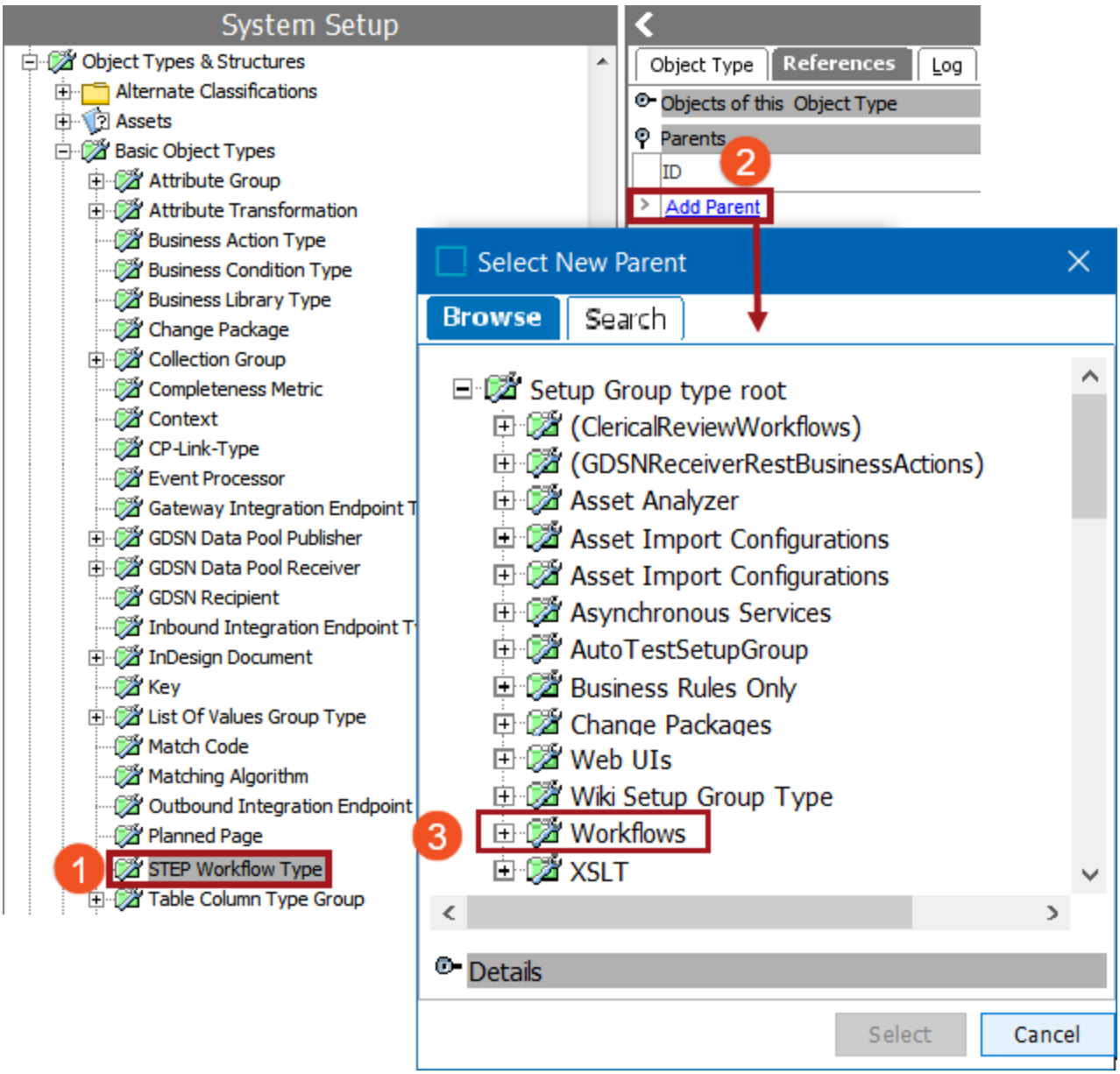
2. Right-click on **Setup Group type root** under **Object Types & Structures** to create a new **New Object Type** from the context menu.



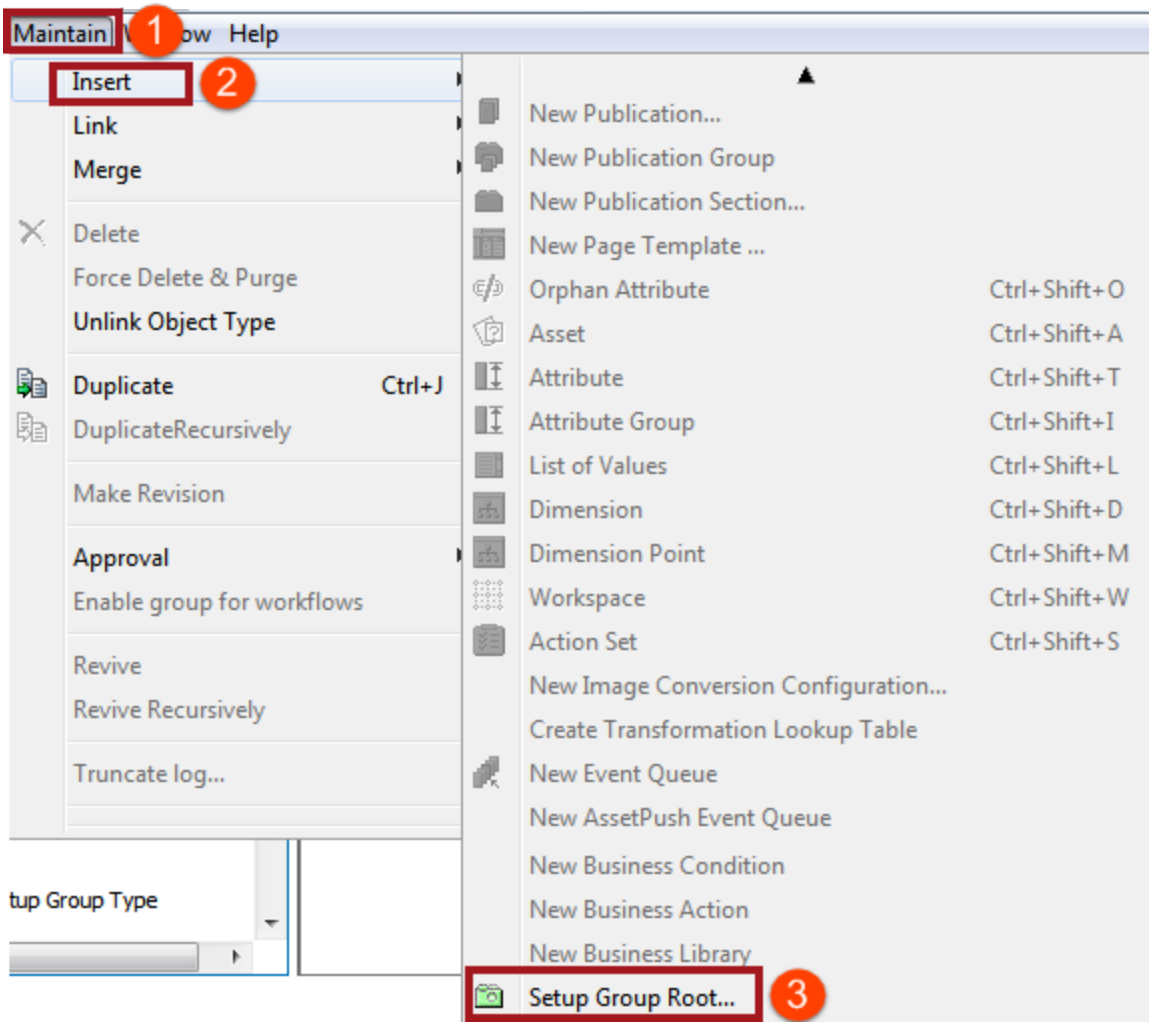
3. Enter an ID and a Name, and click Create.



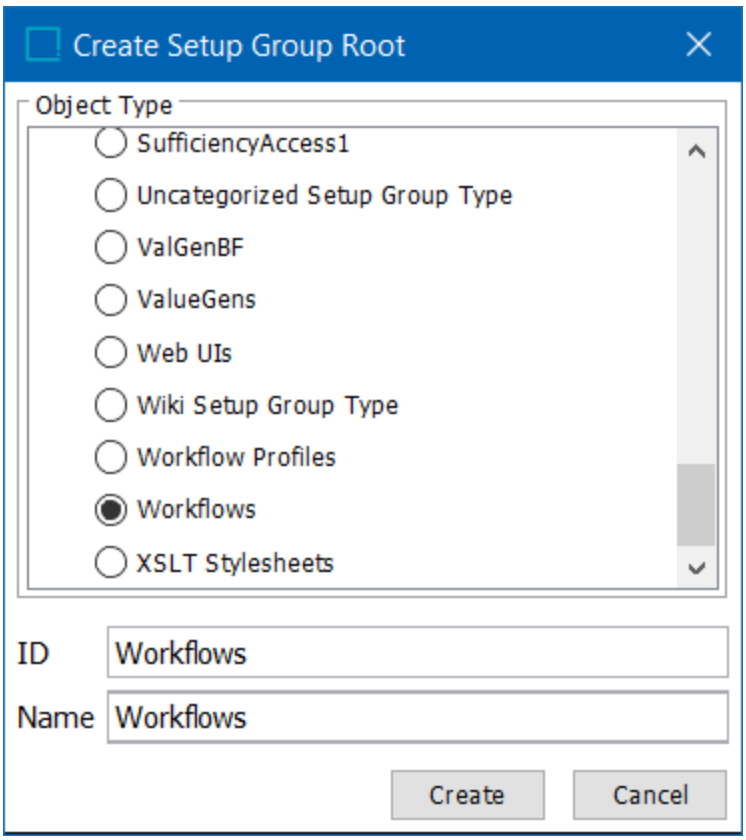
4. Return to the **STEP Workflow Type** node in **Object Types & Structures** > **Basic Object Types**. In the corresponding editor, select the References tab and select **Add Parent** to link this to the folder created for workflows under **Setup Group type root**. This enables workflow objects to be created as children to the Workflows setup group root.



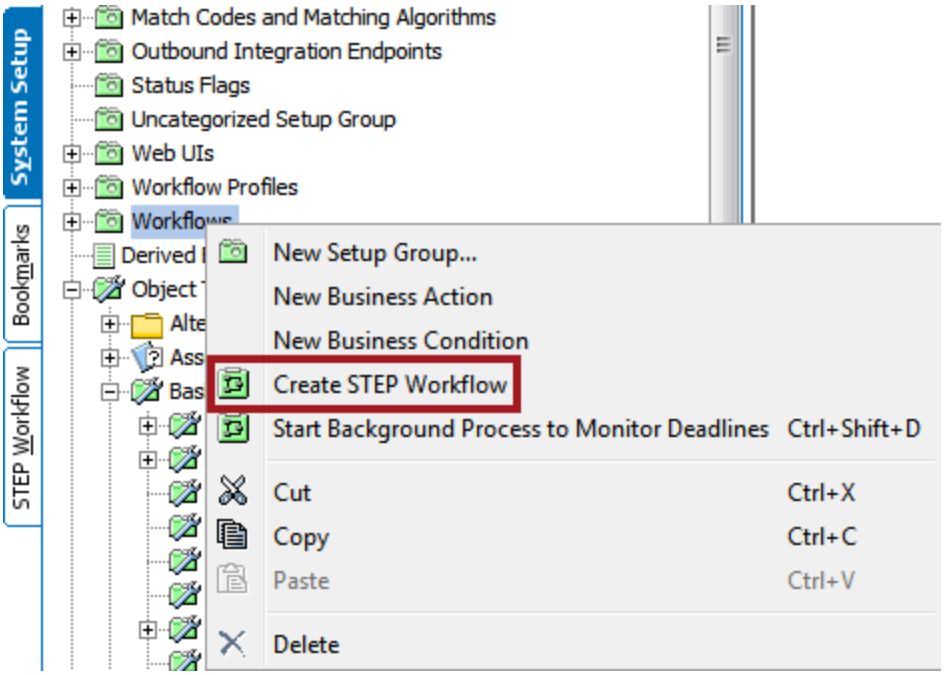
5. To create the parent node for workflow instances to be created under, go to the **Maintain** menu, navigate to **Insert**, and select **Setup Group Root**.



6. Selection of Setup Group Root opens the 'Create Setup Group Root' dialog. Select the workflow object type that was created in Step 3, enter an ID and Name, and click **Create**.



7. This creates a folder in System Setup that a user can then create individual workflows under by right-clicking and selecting **Create STEP Workflow**.

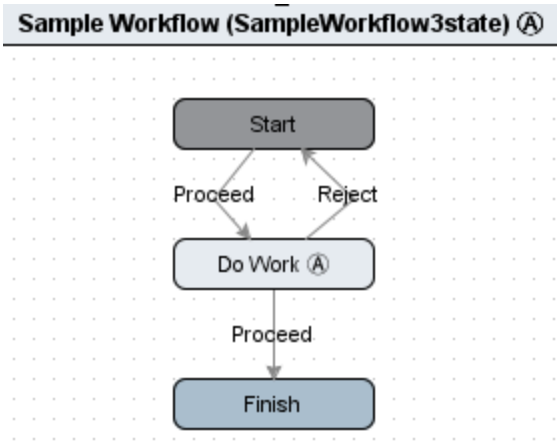


Creating a Workflow

This topic describes how to create a basic workflow, which requires the following steps:

1. Create a new workflow
2. Create states in a workflow
3. Create transitions between states
4. Add events to transitions
5. Set assignees
6. Set validity
7. Set workflow options

Each of the steps is described in detail below, guiding users through the creation of a simple workflow, as shown below.



In addition, a new workflow can be created via duplication of an existing workflow, so that topic is also covered in this section.

Prerequisites

In order to create a workflow, the system must first have a place established to house the workflow. This setup only needs to be completed once on any system, and is likely to have already been completed. However, if the System Setup tab does not have a folder that allows workflow children (as shown in the Step 1: Create a New Workflow section below), then this initial configuration needs to be completed before any workflows can be created. The steps for the configuration are detailed in the **Initial Setup for Workflows** topic.

Preconsiderations

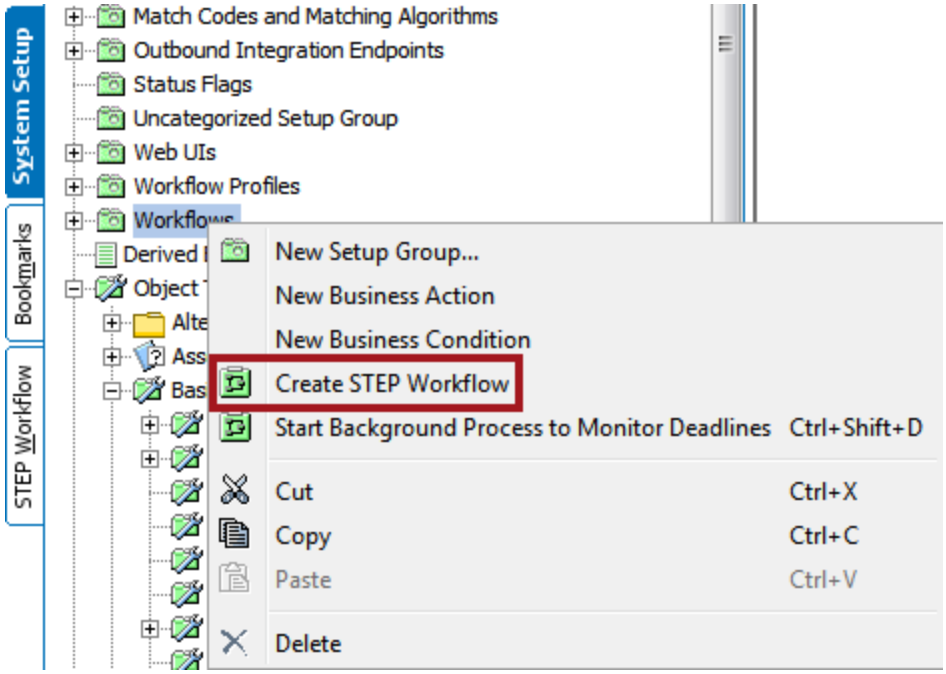
Steps 1 - 3, and step 5, of workflow creation must all be completed in a single session in order to save the workflow configuration. Therefore, care should be taken during initial configuration to ensure that the steps can be completed and work is not lost.

Each of the items below must be true in order to save a workflow configuration:

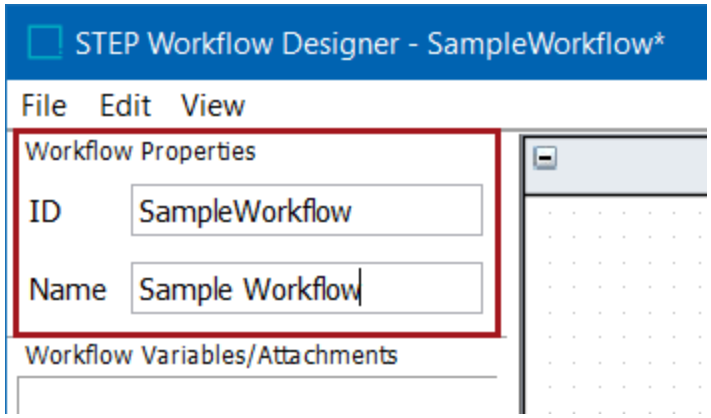
- A unique ID must be specified for the workflow
- There must be an Initial and a Final state on the outer level
- All states must be reachable (e.g., there must be at least one incoming transition for each state)
- State IDs must be unique within the workflow
- All states must have an associated assignee, or a default assignee must be assigned to the workflow

Step 1: Create a New Workflow

1. Right-click the folder established to house all workflows, and select **Create STEP Workflow**.



2. The **STEP Workflow Designer** will open. Additional information about the designer can be found in the **STEP Workflow Designer** topic. For now, enter an ID and Name for the workflow.



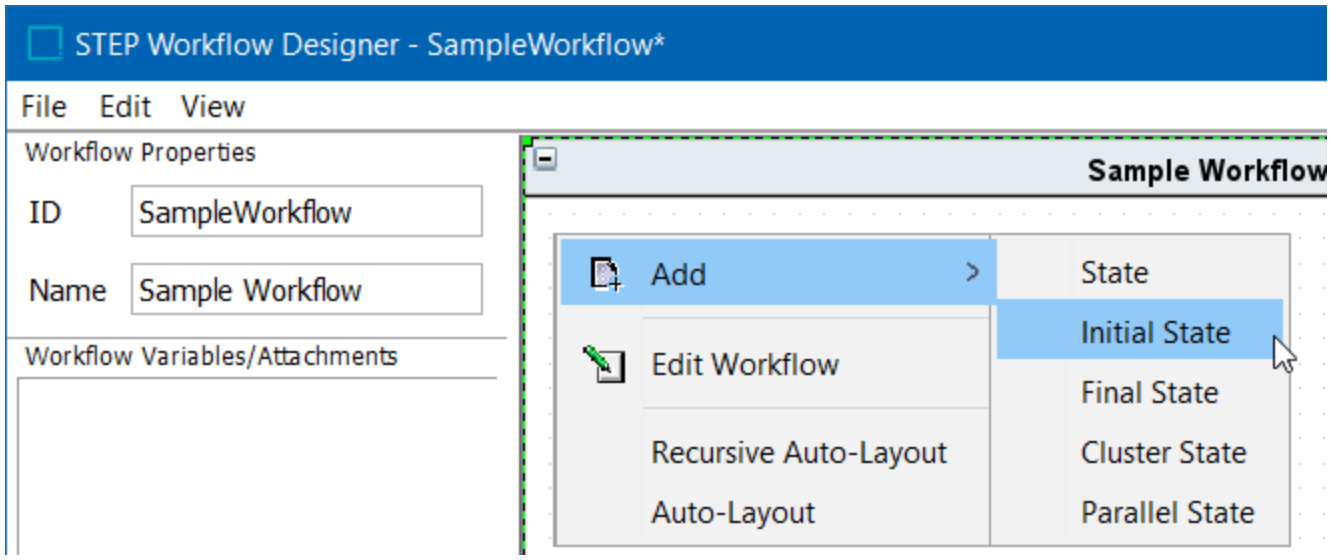
At this point, a 'container' has been created, to which states and transitions must be added to build the actual workflow.

Step 2: Create States in a Workflow

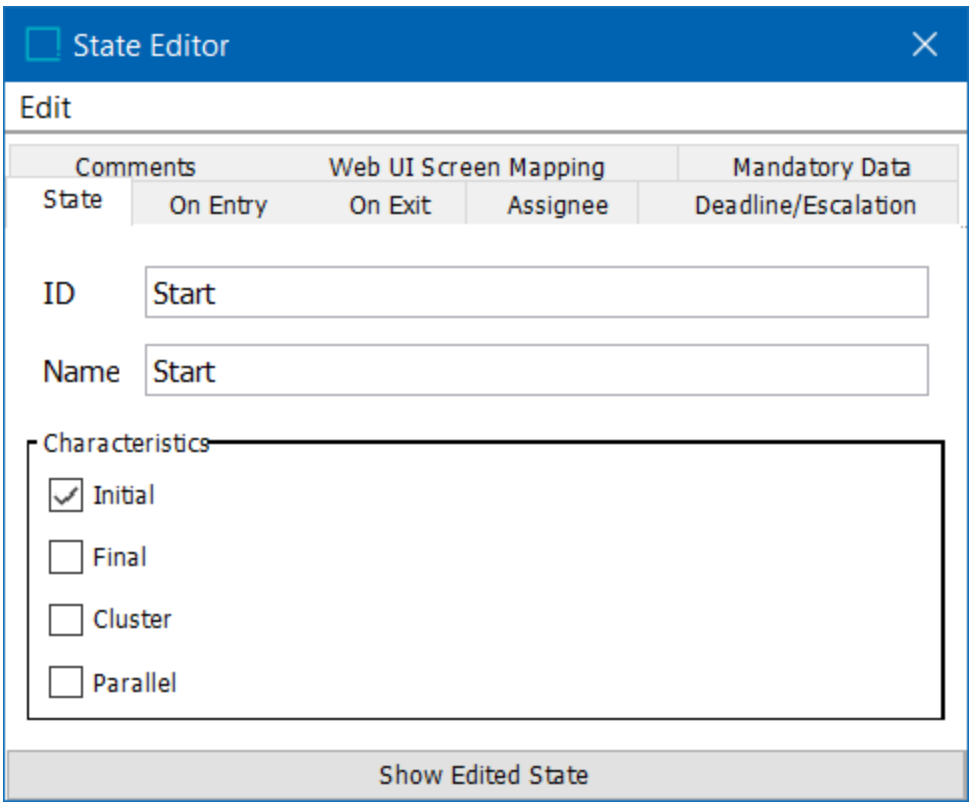
A state represents an assignment to be completed, whether by a human or a system. When a new workflow is created, it must have a single initial state at the outermost level, which is where any task entering the workflow will begin. In addition, a single final state must be present at the outermost level to indicate completion of the workflow. The indication of 'initial' or 'final' for a state is considered a characteristic of the state. States can also have 'parallel' or 'cluster' characteristics, which are required when workflows have parallel states and/or when a set of states should be grouped together within a workflow (comparable to a mini-workflow within the overall flow).

For this example, we will create a basic workflow using three states: an initial state, a standard state, and a final state. However, further descriptions of state characteristics, including more complex examples using parallel and cluster states can be found in the **State Characteristics in Workflows** topic of the **Advanced Workflow Topics** documentation.

1. To create the **Initial State**, right-click the canvas area and select **Add** > **Initial State** from the menu. The **Initial State** is the first state that any object started in the workflow is placed into.



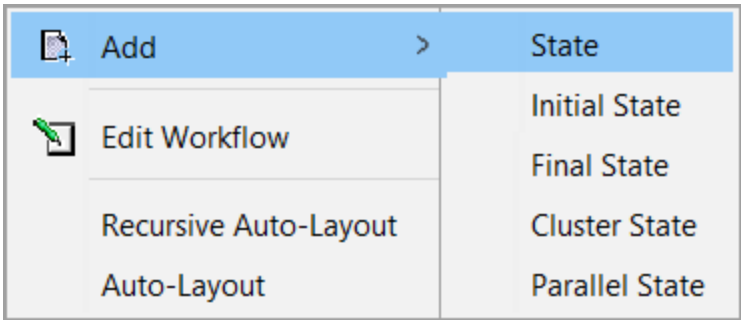
2. In the State Editor that appears, type the desired **ID** and **Name** of the state. Make sure that **Initial** under **Characteristics** is checked, and close the dialog box by clicking the 'X' in the top right corner.



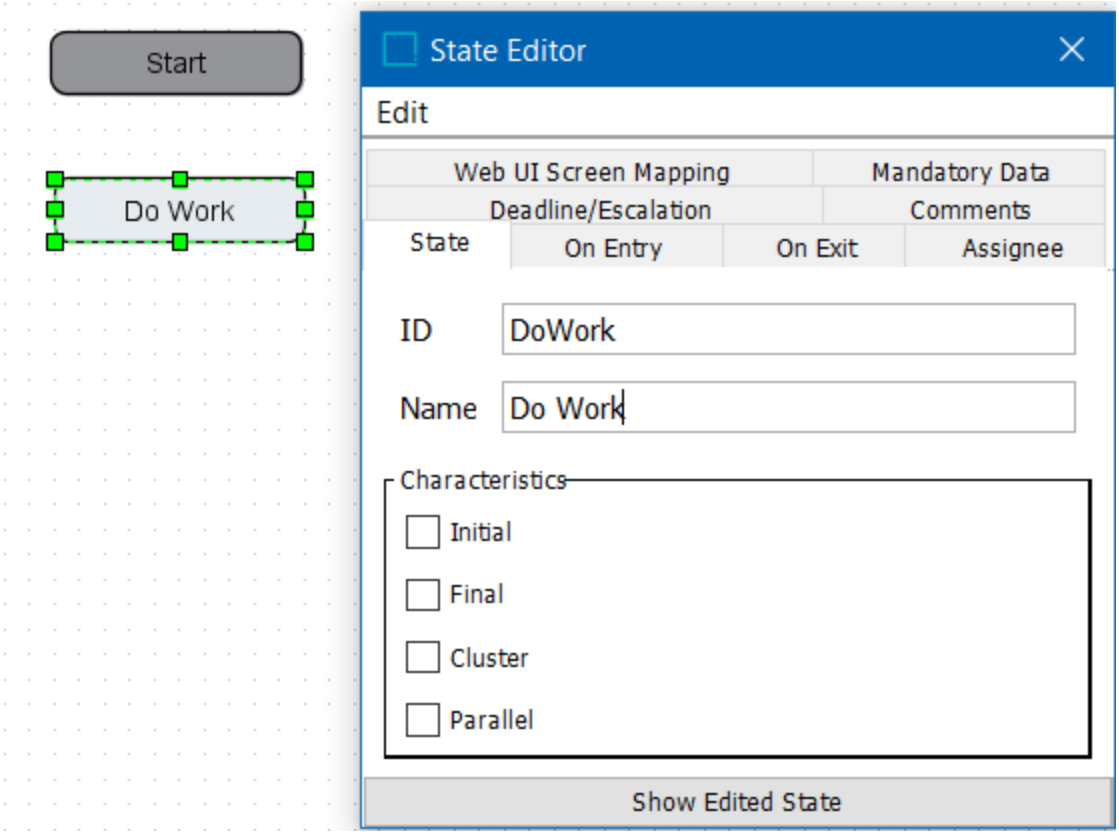
Note that IDs can be revised as needed before the workflow is saved. However, because workflow business rules are tied to state IDs, all state IDs are locked upon saving of the workflow.

State names are editable at any time, and can also be localized, allowing users to see differing names for the same state based on locale. More information on this can be found in the **Localizing Workflow State Names and Events** section of the **Workflows** documentation.

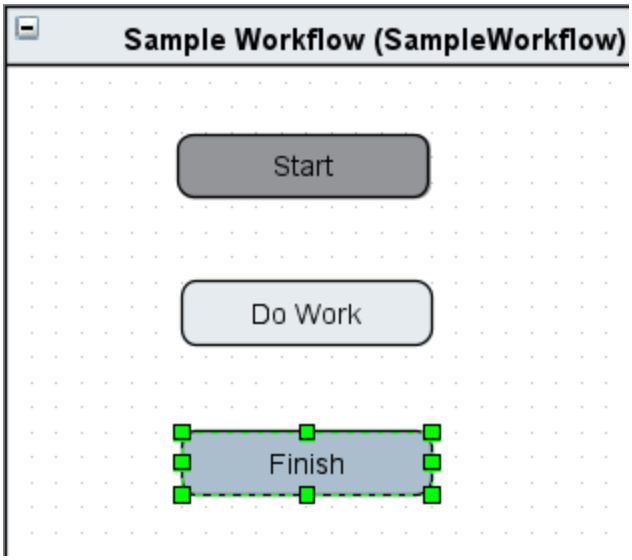
3. Create a second state by right-clicking the canvas and selecting **Add** > **State**.



No **Characteristics** need to be checked in the State Editor dialog box for states in between the **Initial** and **Final States**.



4. For the last state, right-click the canvas area and select **Add** > **Final State**. In the State Editor box, fill in the ID and Name accordingly, making sure that **Final** is checked under **Characteristics**. Close the dialog. The workflow should now appear as pictured below.

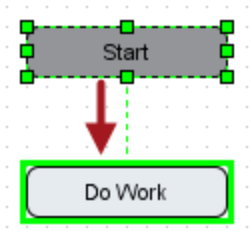


You have now created a basic workflow with three states. However, in order for tasks to be able to move through the workflow, transitions must be added between the states.

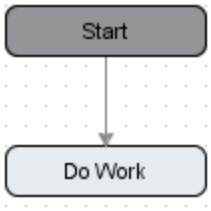
Step 3: Create Transitions Between States

Transitions are needed between states in order for the workflow to be valid. Each state (with the exception of final states in the outer level) should have at least one transition to allow tasks to proceed through the workflow. States may have more than one transition, meaning there are multiple options for exiting the state. For example, users may need the option to move a task to a subsequent state in the workflow, as well as returning it to a previous state (e.g., to proceed or reject).

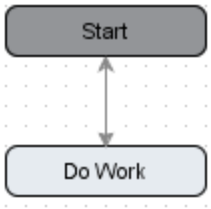
1. Hover over the center of the **Initial State** so that the cursor changes into a hand icon and a green box appears around the state. Click and drag to the center of the next desired state until a green box encircles that state.



   Release the mouse click to create the transition, which is then displayed as an arrow between the states.

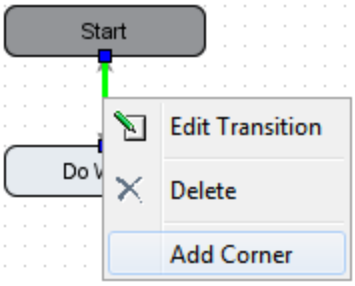


2. Repeat step 1, but going in the opposite direction (e.g., click the second state and drag to the first).

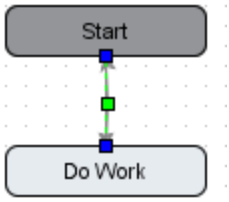


   Now there are two transitions going in opposite directions in the workflow. This allows objects to progress forward in the workflow if approved or to be rejected and sent back to a previous state if needed.

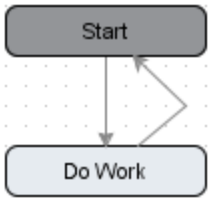
3. Though they appear as a single arrow at this point, each transition is actually unique, which can be seen by adding corners to each transition. To do this, right-click on the arrow and select **Add Corner**.



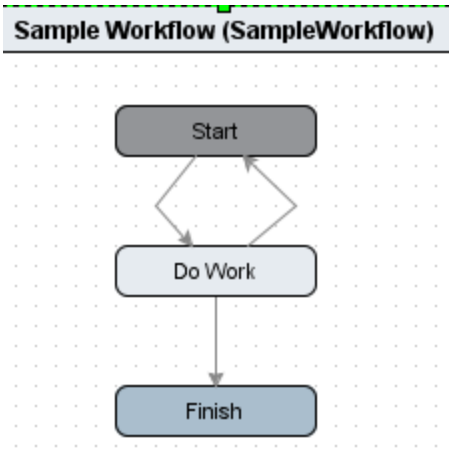
A green box will appear on the transition.



Click and drag the box to the right or left.



4. Repeat the above steps to add a corner to the other existing transition, and to add a new transition between the second and third states. When complete, your workflow should appear as shown below.



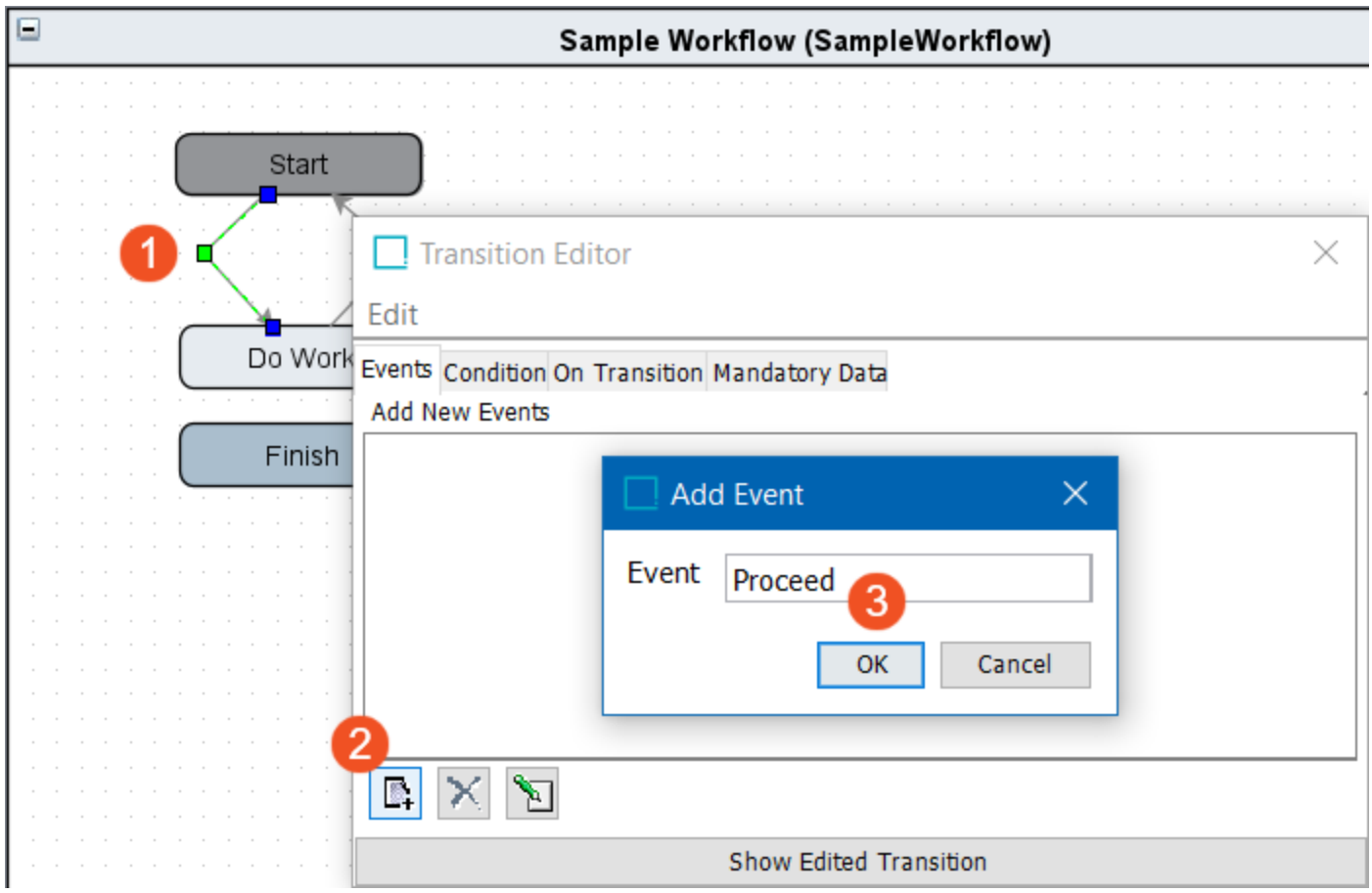
Step 4: Add Events to Transitions

While not strictly necessary, adding events to transitions can increase usability of workflows as the event names are displayed to the end users. When a meaningful event name is applied, it provides an indication to the user of what is occurring when they select to Submit a task in the workflow. When multiple transitions are present from a single state, it is especially useful to have events applied so that a user can choose which selection to make (e.g., Proceed vs. Reject).

Events also enable advanced workflow options, such as applying business rules and logic to specific events. More information on advanced functions is available in the **Advanced Workflow Topics** section of the documentation.

To add an event to a transition, follow the below steps.

1. Double-click the transition, select the **Events** tab in the Transition Editor dialog that appears, and then click the **Add Event** icon button. Name the event accordingly and click **OK**.

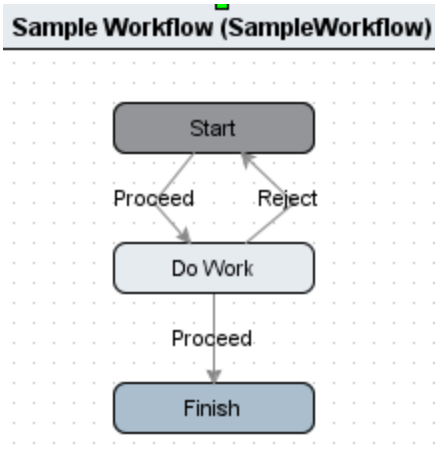


Note that if a space is put in the event name, the Designer will automatically replace the space with an underscore within the editor. However, when the event is displayed to end users in the workbench or Web UI, the underscores will render as spaces.

2. Additional events can be added to the same transition, which can be useful when adding business rules to a workflow. However, for this simple example, we will use only one event per transition so the Transition Editor

should be closed by clicking the X in the upper right corner. Once closed, event names can be dragged on the canvas to improve visibility if needed.

3. Repeat the above steps to add an event to each remaining transition, resulting in a workflow comparable to the one shown below.



Note that multiple events within the workflow may have the same name.

You have now created a basic workflow, but in order to make it valid (which is required to enable saving of your work), an assignee must also be specified for the workflow.

Step 5: Set Assignees

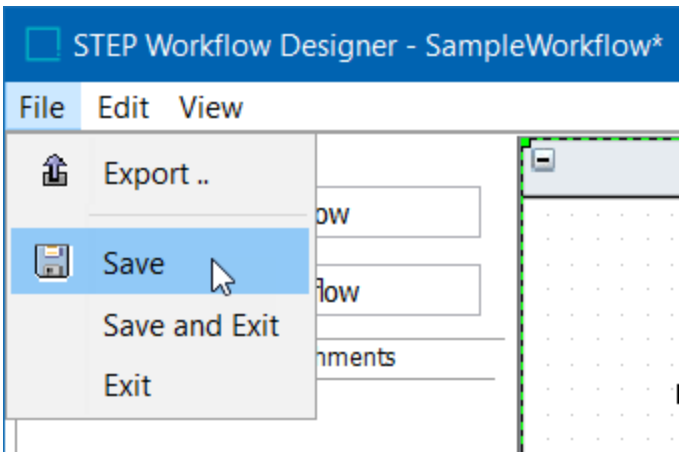
Each state in a workflow requires an assignee to be set, regardless of whether the state represents a human task or not. The object-in-state relation must always be assigned to a STEP user or user group. This can be done one of two ways: individual assignees to each state or a single assignee to the overall workflow. If states do not have an assignee defined locally, they will by default always inherit the assignee defined by the workflow.

As the primary purpose of workflows is to assign tasks to varying actors in a controlled manner, each workflow state that requires human interaction will generally have a different assignee (rather than using the workflow default). In contrast, states that represent a system action or an automated state will often use the workflow default, which can be set to an administrator user group.

Note: Users that do not have the STEP Workflow Administrator privilege will only be allowed to see tasks assigned to themselves or a group they are member of.

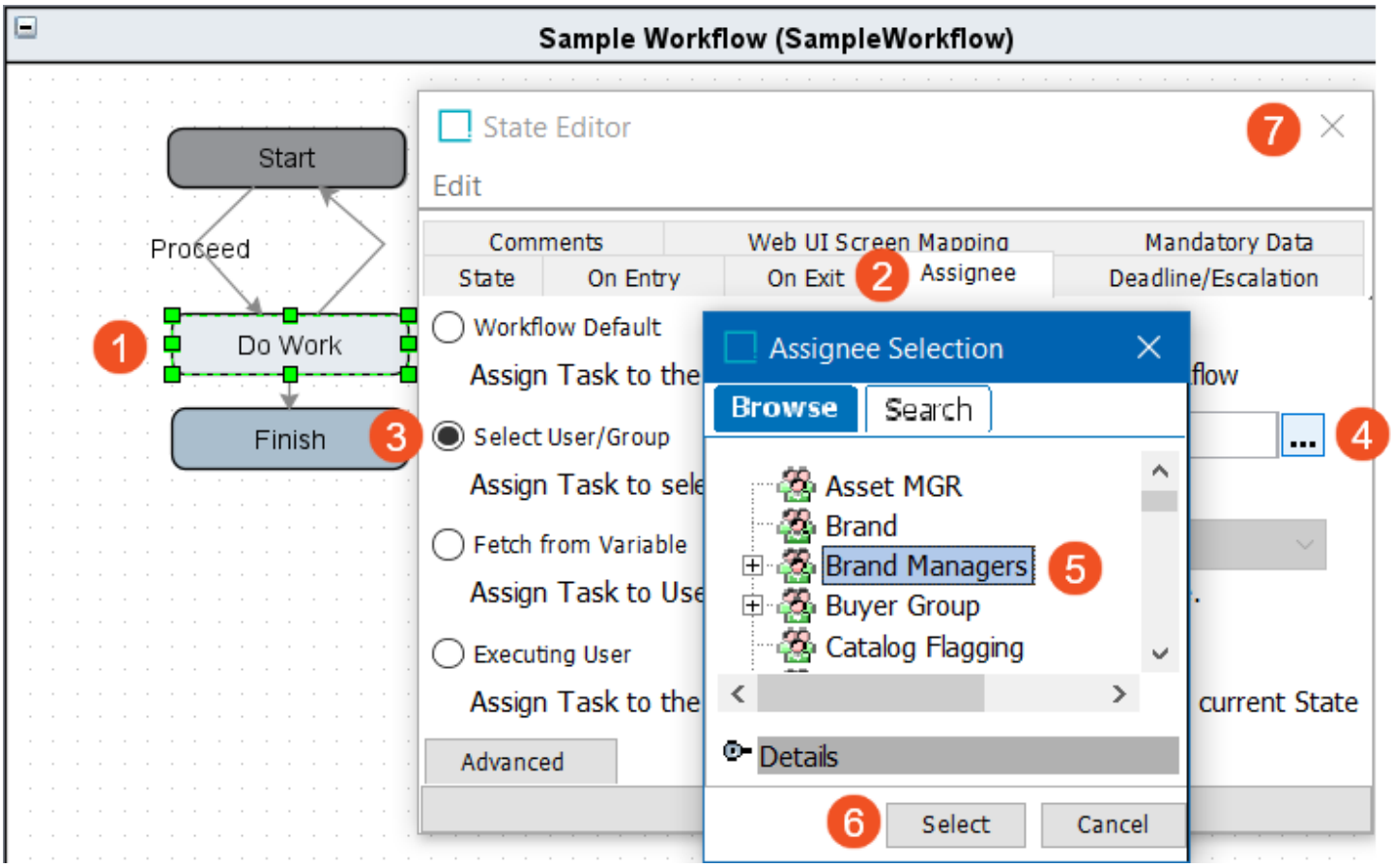
There are several options for setting assignees in a workflow, including to set assignee dynamically based on information on the object or previous actions taken in the workflow. For this example, the basic functions of specifying a user or group assignee are described. More advanced assignee options are described in the **Assignees in Workflows** section of the **Advanced Workflow Topics** documentation.

Once assignees have been set, the workflow should be saved:



Set an Assignee for a State

All states with user interaction should have an assignee specified. To set the assignee locally for a state, double-click the state, select the **Assignee** tab in the State Editor dialog that appears, click the **Select User / Group** radio button, click the ellipsis button (...) to open the Assignee Selection dialog, select a user or user group, click **Select** to close the selection dialog, then click the X in the upper right corner to close the State Editor dialog.

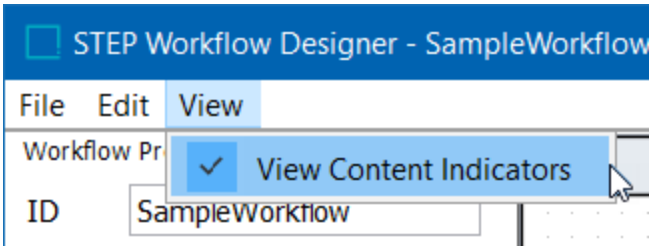


When a user group is selected, all users in that group can access tasks in the state. When a specific user is selected, only that user (or a workflow administrator) can access tasks in the state. It is most common to assign

states to user groups rather than to individual users so that tasks can continue to be processed even if a specific user is unavailable, for example when sick or on holiday.

When a state has an assignee, the letter 'A' with a circle around it will appear in the state: 

If the assignee indicator does not appear, go to the View menu in the STEP Workflow Designer and check the **View Content Indicators** option.

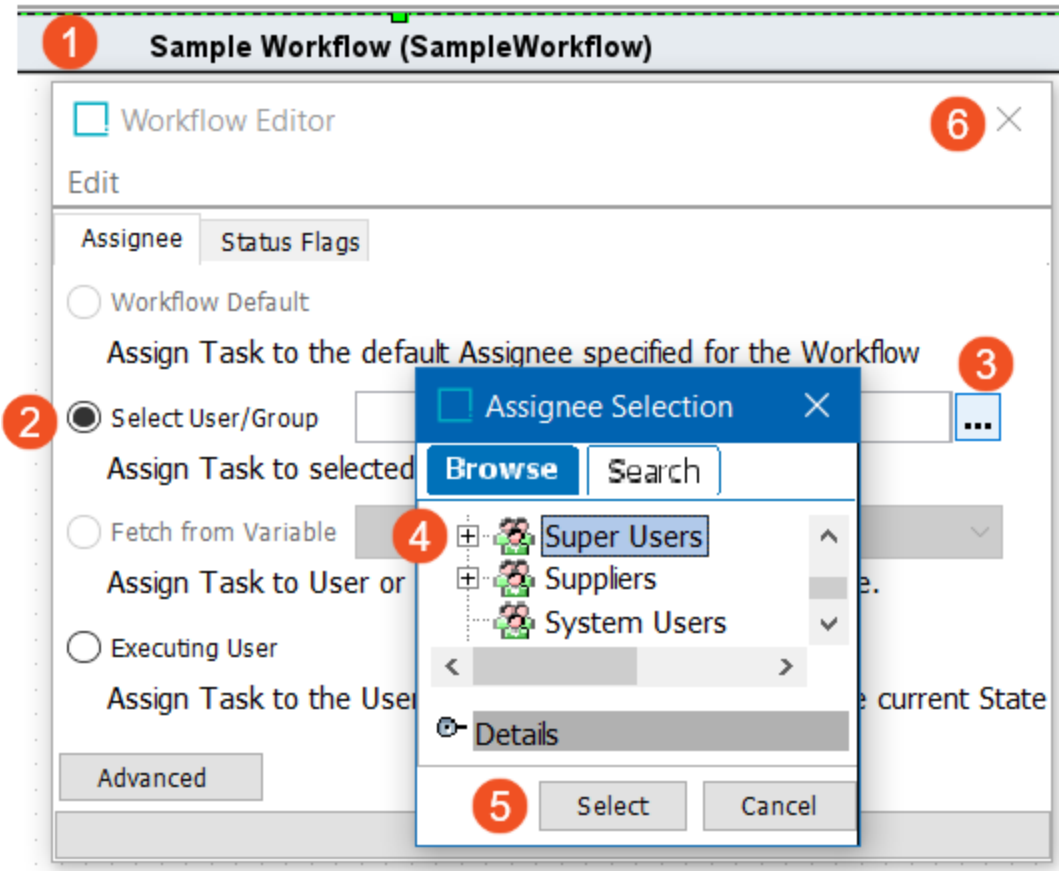


The above actions can be repeated until an assignee has been designated for each state in the workflow. Alternatively, an assignee can be set for the complete workflow. When each state has an assignee (via individual assignment or the workflow default), the workflow can be saved via the Save option on the File menu.

Set an Assignee for the Workflow

In order to save a workflow, all states must have an assignee specified. All states that do not have an assignee explicitly specified will inherit the workflow assignee. Therefore, it is often helpful to have a default assignee specified to ensure that the workflow can be saved at any point. In addition, it is common to use the workflow assignee for states that are automated and/or the work occurring in the state is done by the system rather than a user as in these cases the assignee is not a meaningful designation. Typically the default workflow assignee will be an administrator user or group.

To set the default assignee for a workflow, double-click the title bar of the workflow, click the **Select User / Group** radio button, click the ellipsis button (...) to open the Assignee Selection dialog, select a user or user group, click **Select** to close the selection dialog, then click the **X** in the upper right corner to close the State Editor dialog.

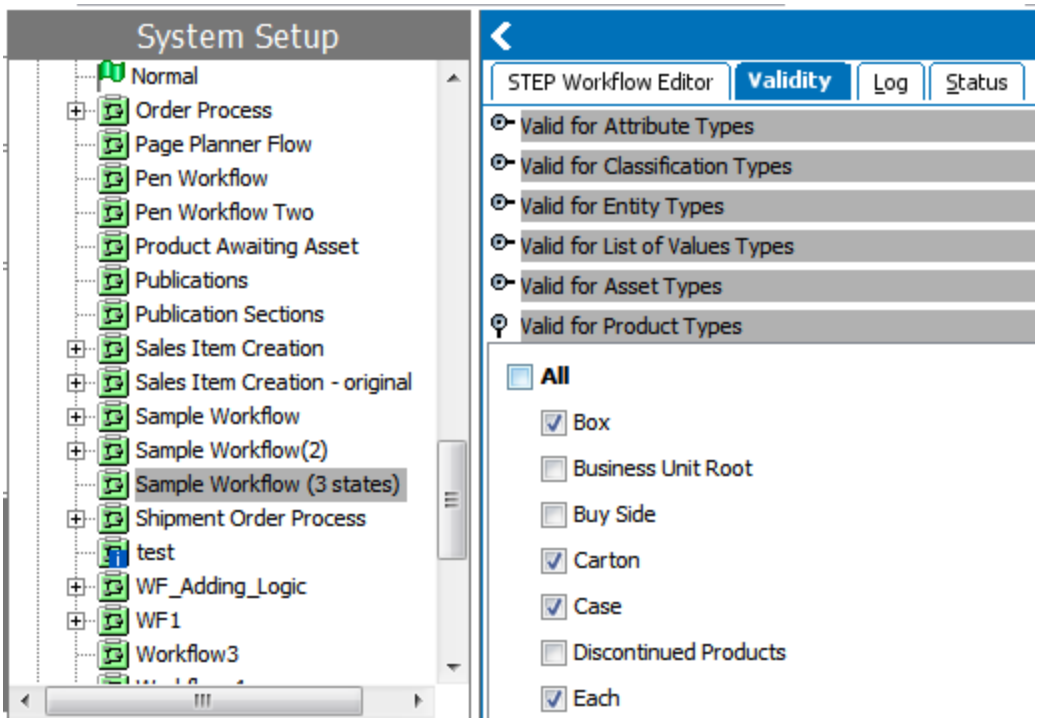


When each state has an assignee (via individual assignment or the workflow default), the workflow can be saved via File > Save.

Step 6: Set Validity

Before a workflow is fully functional, it must be made valid for the object type(s) that will use it. For example, if a workflow was made specifically to review and approve packaging objects and data, it might be made valid for Each, Box, Carton, and Case.

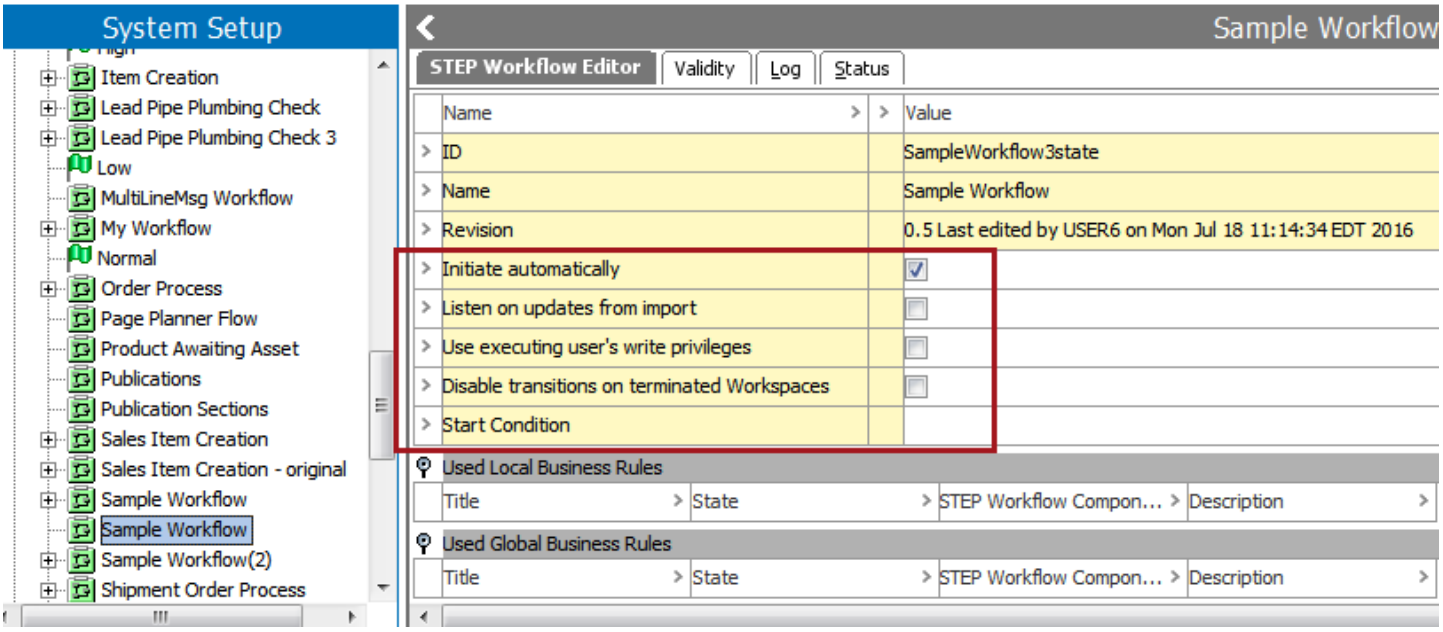
To set validity, select the workflow from the **System Setup** tab, click the **Validity** tab, and check the appropriate object type(s). Any number of object types can be selected within the same flipper.



Note: Workflows can be made valid for several object types, but only types of the same 'super type.' As an example, you could make a workflow valid for the object types 'Shipping Address' and 'Main Address' of the super type 'Entities', but not also for 'Building Photo' of the super type 'Assets.'

Step 7: Set Workflow Options

The final step in creating a workflow is to set the options available on the STEP Workflow Editor tab. These settings are not required to be changed, but they provide valuable options for governing the behavior of the workflow and should be considered as part of any workflow design.



Initiate automatically - If checked, objects of the specified validity for the workflow are automatically initiated into the workflow upon creation. This option is only applicable for object types for which instances of the objects can exist in the Tree (e.g., products, classifications, assets, entities, publications, publication sections, or planned pages, but not for attributes or lists of values). This is useful for onboarding / creation workflows that are intended to process new objects as the objects are automatically placed in the workflow upon creation, without any manual initiation required.

Listen on updates from Import - If checked, the system will listen for updates on objects of the type specified by the validity of the workflow and take action as needed. Specifically, an object can be transitioned from one state to another upon update of the object via import. This is especially useful when completion of a workflow requires information from another source system. As soon as that information is received, the task can be automatically sent to the next state in the workflow. This function is only available for assets, classifications, entities, and products and requires additional configuration to enable. Complete configuration is described in the **Triggering Workflow Events from Imports** section of the **Advanced Workflows Topics** documentation.

Use executing user's write privileges - Option related to the execution of business rules in the workflow. If selected, a user triggering the execution of a business rule will be prevented from making any changes or accessing any data in the system via the business rule that he / she is not normally privileged to. For example, if the user is not privileged to change a certain attribute value on an object, and the business action is configured to make such a change, a privilege exception will be thrown and the action will not be carried out. Caution should be used with this option as it can prevent desired business rules from being carried out and/or result in errors that the business rule should be written to handle gracefully. If the option is not checked, business rule are run with full access to carry out all indicated rules.

Disable transitions on terminated Workspaces - If selected, it will not be possible to perform transitions in terminated workspaces (e.g., the Approved workspace). Typically this option should be checked as it will prevent a user from submitting objects while in the Approved workspace. This is important as data cannot be modified in the Approved workspace, submission of a task to a state often involves business actions, and workflow actions are executed from the current workspace. Therefore, if a user is viewing data in the Approved workspace and

attempts to submit an object from a state where submission involves the use of a business action, the user will receive an error. Checking this option makes the submission unavailable, and therefore prevents the user from receiving an unnecessary error.

Start Condition - Option available to specify a business condition that needs to be fulfilled before the workflow can start. For example, if some basic data is required to be populated prior to initiation, a condition can be selected to ensure that the required data is present. If the condition is not met (e.g., does not evaluate to true), the object will not be allowed to enter the workflow.

Important: If users will be working with tasks within the workbench (rather than Web UI), views and mappings should also be configured following creation of the workflow. Instructions for this configuration are available in the Views and Mappings section of the Getting Started with STEP Workflows material.

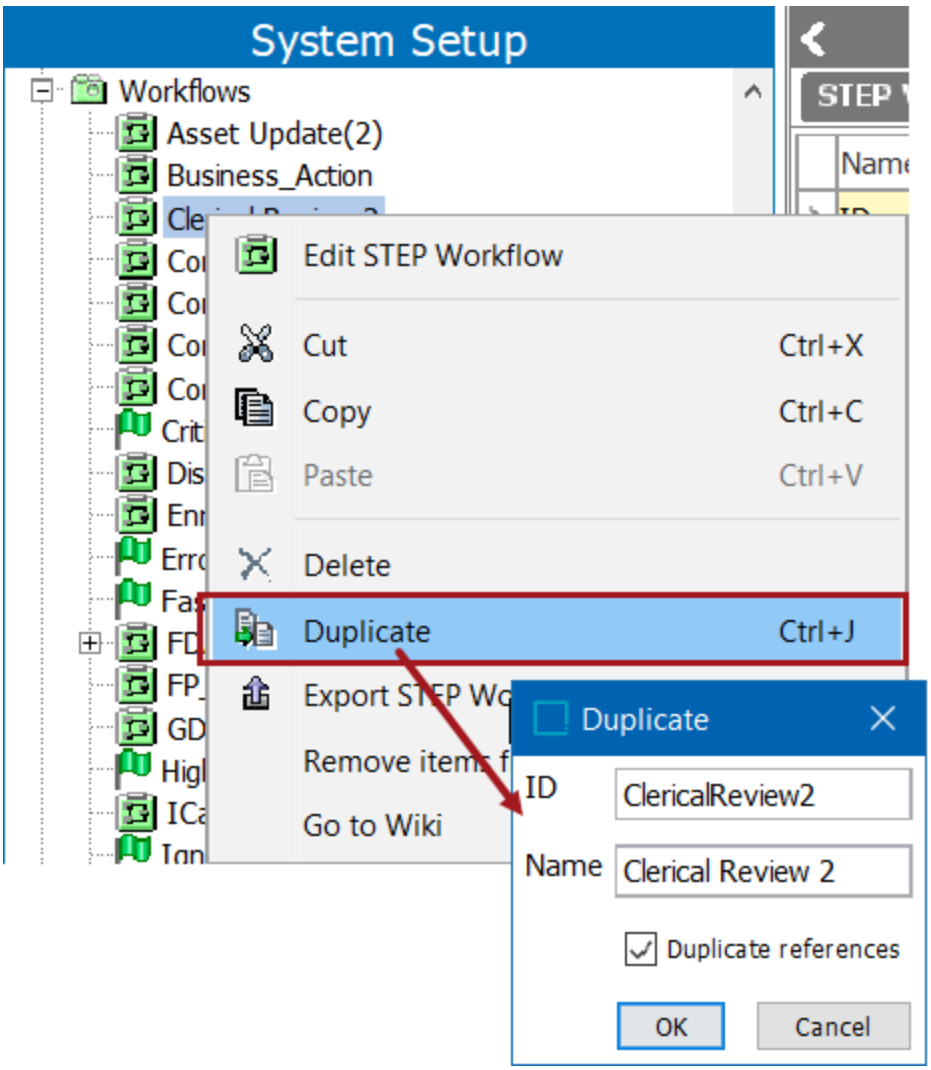
Duplicate a Workflow

Another way to create a workflow is to duplicate an already existing workflow. While there may be a number of reasons for duplicating an already existing workflow, some of the reasons may include:

- The need to make changes to a workflow already in use. In duplicating the workflow, it is possible to make changes to the new workflow and test these changes while still leaving the old workflow intact.
- The need to create a new workflow similar to an existing workflow.

Duplicating a workflow will result in the basic workflow structure and associated business rules being copied. However, the global workflow options (Step 7 in the previous material) will not be copied and should be set manually on the duplicated workflow. In addition, workflow-specific privileges are not carried over in a duplication.

To duplicate an existing workflow, go to System Setup and right-click on the desired workflow for duplication. Select **Duplicate**. In the dialog that appears, populate the ID and Name, and click the OK button. The 'Duplicate references' option is not applicable for workflows and can be left checked or unchecked.


System Setup
Workflows
Asset Update(2)
Business_Action
Edit STEP Workflow
Cut Ctrl+X
Copy Ctrl+C
Paste Ctrl+V
Delete
Duplicate Ctrl+J
Go to Wiki
Duplicate
ID
ClericalReview2
Name
Clerical Review 2
Duplicate references
OK
Cancel

STEP Workflow Designer

This topic provides an overview of the STEP Workflow Designer, including descriptions of the various sub-editors and visual indicators within the designer. It is intended to orient you with the interface and provide general information for where to accomplish things in the designer, rather than to complete any specific task itself. When applicable, links to more detailed task-oriented materials are provided.

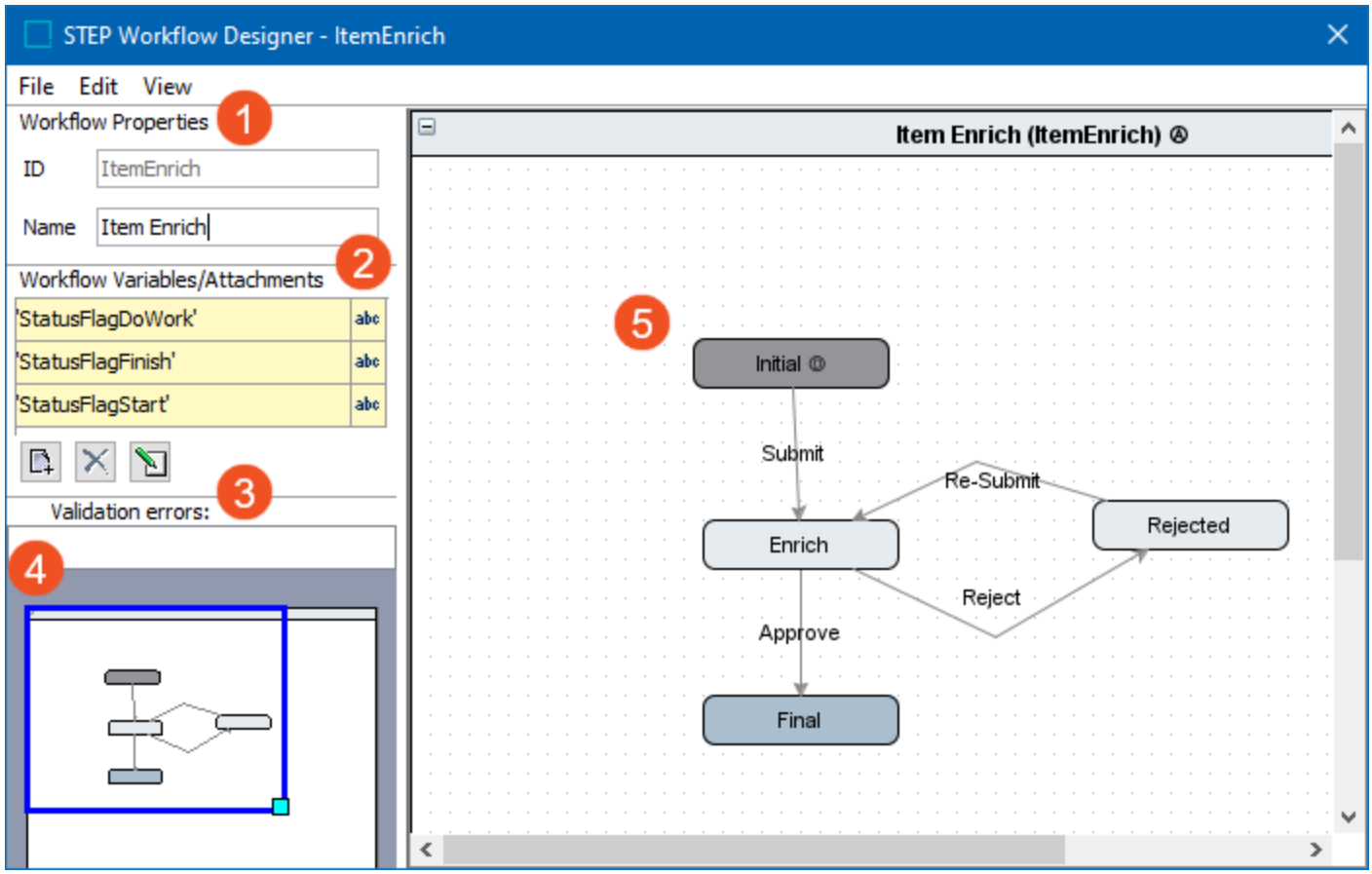
Prerequisites

It is expected that the reader is familiar with the **Creating a Workflow** topic.

Overview

The STEP Workflow Designer has the following primary areas:

1. Properties panel for setting **ID** and **Name**
2. Area for defining **Workflow Variables/Attachments**
3. Area for displaying **Validation errors** and warnings
4. Overview / zoom area
5. Canvas where States and Transitions are defined

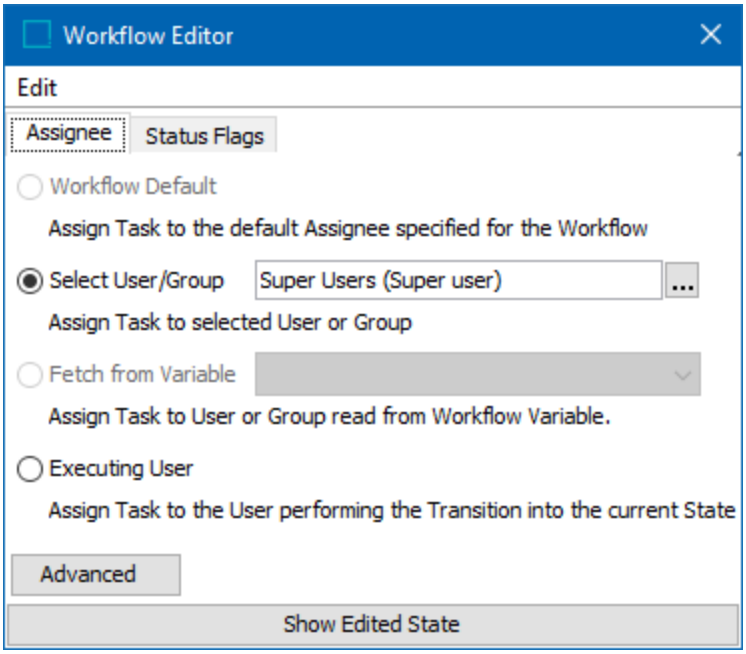


Dialogs within STEP Workflow Designer

There are a number of dialogs that have many additional options and features that aid users in editing workflows. Each dialog and the general functions of the included tabs are described below.

Workflow Editor

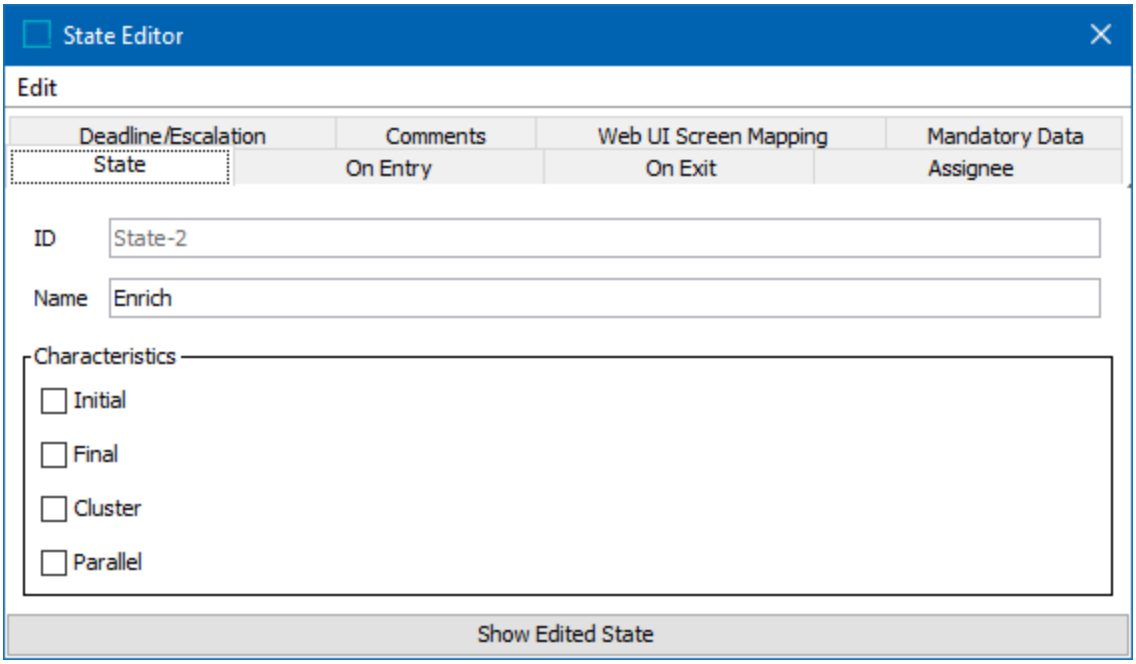
To edit in the Workflow Editor, double click the canvas region. The workflow editor is used to make changes that are global for the workflow, including setting the default assignee for the workflow and enabling status flags for the workflow.



Tab	Available Functions
Assignee Tab	Modify the assignee for the entire workflow. See **Workflow State Editor Assignee Tab Options**.
Status Flags Tab	Status Flags can be added and removed. See **Status Flags**.

State Editor

The State Editor is launched when the user right-clicks on the canvas to add a state, or when a user double-clicks on an existing state. This editor enables users to make changes specific to a particular state.

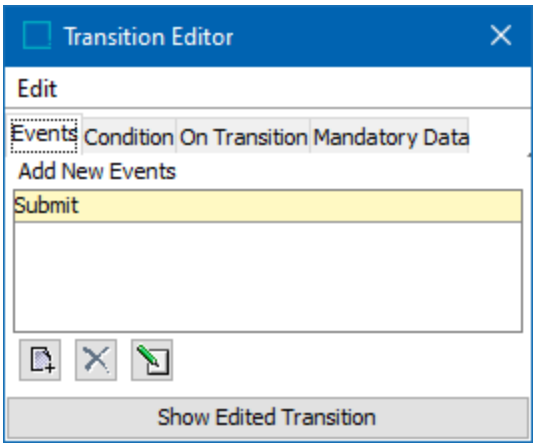


Tab	Description
Assignee	Assignees can be added / removed / modified per state. See **Assignees in Workflows**.
Comments	Comments can be added / removed / modified. Comments have no explicit function and are not viewable outside of the designer. However, they can be useful to make notes for designers regarding a particular state (e.g., intended function, considerations when modifying, etc).
Deadline/Escalation	Set deadlines and add / modify / delete associated escalation business rules. See **Deadlines and Escalations**.
Mandatory Data tab	Set attributes, attribute groups, and/or references that are required to be populated before a task can leave the state. See **Mandatory Attributes and References in Workflows**.
On Entry	Add / modify / delete business actions that are run on entry to the state. See **Business Rules in Workflows**.
On Exit	Add / modify / delete business actions that are run on exit from the state. See **Business Rules and Workflows**.

Tab	Description
State	Change the name or characteristics of the state. See **State Characteristics in Workflows**.
Web UI Screen Mapping	Provides a link to the Web UI to map the state to a specific Web UI screen. From the dropdown, select the Web UI that should have access to the workflow, and click the Edit button. This will launch the login page of the Web UI. The user must have privileges to edit Web UI configurations and if so, logging in will open the designer to the Screen Mapping properties with conditions to map to the workflow state pre-populated. The user can then select an existing screen to map the state to. More information on mapping Web UI screens can be found in the **Main Properties** topic of the **Web User Interfaces** documentation.

Transition Editor

The Transition Editor is launched when the user double-clicks on an existing transition. This editor enables users to make changes specific to a particular transition.



Tab	Description
Condition	Add / remove / modify business conditions on the transition. See **Business Rules and Workflows**.
Events	Add / remove / modify events on the transition. See **Auto Routing Tasks in a Workflow**.
Mandatory Data	Set attributes, attribute groups, and/or references that are required to be populated before a task can take the transition. See **Mandatory Attributes and References in Workflows**.

Tab	Description
On Transition	Add / remove / modify business actions on the transition. See **Business Rules and Workflows**.

Visual Indicators in STEP Workflows

Several visual indicators are present to allow users to quickly gather information about the workflow without having to open a particular state or transition.

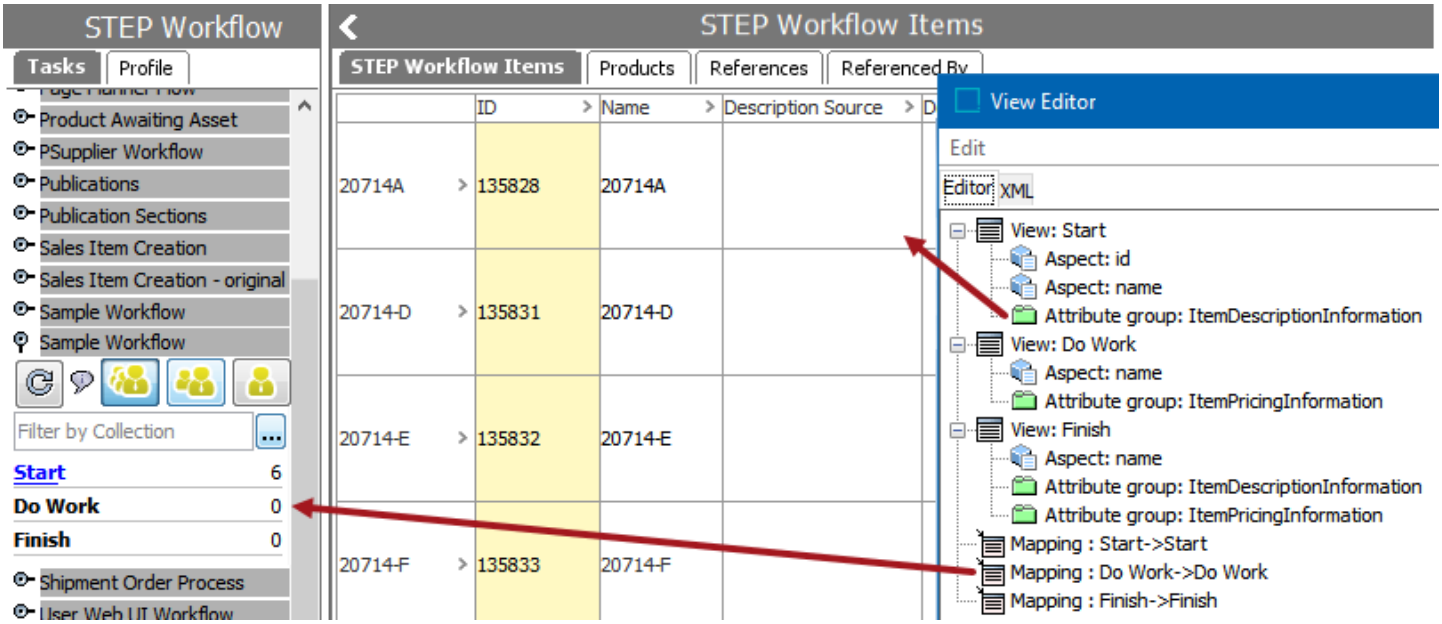
Indicator	Description
'A'	A assignee has been specified for the workflow or state. See **Assignees in Workflows**.
'D'	A deadline has been selected for the state. See **Deadlines and Escalations**.
'I'	An 'On Entry' action is present on the state. It may help to remember this as 'I' for 'In' (e.g., into the state). See **Business Rules and Workflows**.
'O'	An 'On Exit' action is present on the state. It may help to remember this as 'O' for 'Out' (e.g., out of the state). See **Business Rules and Workflows**.
'P'	Specific user privileges have been granted for the state. See **Editing Privilege Rules**.
Dotted transition line	A condition has been placed on the transition. See **Business Rules and Workflows**.
Rounded end on transition	An action has been placed on the transition. See **Business Rules and Workflows**.
Blue color	The state has the Final characteristic. See **State Characteristics in Workflows**.
Dark gray color	The state has the Initial characteristic. See **State Characteristics in Workflows**.
Light blue / gray color	The state has the Cluster characteristic, or no characteristics (e.g., standard state). See **State Characteristics in Workflows**.

Views and Mappings for Workflows

While not strictly necessary, if users are going to work with workflows in the STEP Workbench, it is generally recommended to create views and mappings for the workflow. This ensures that users have access to the right information to complete their tasks and streamlines the work process. However, users can also use the standard editors available on objects in the Tree if desired, using the Workflow navigator only to submit tasks rather than to review and edit data.

When a workflow state is selected in the STEP Workflow navigator tab, the information that displays is called a view. The act of applying that view to a particular state in a workflow is called mapping. When configuring a workflow it is important to decide what data is visible for objects (views), and define which states to display (mappings). Without views and mappings configured, all states in the workflow are displayed on the STEP Workflow navigator tab (including parallels, clusters, and final states). Additionally, just the ID of the objects in the selected state display in the STEP Workflow Items editor or the Task tab on the object.

Upon mapping views to states, only the states for which a mapping exists will be displayed in the STEP Workflow navigator tab. The configured views will be used in the STEP Workflow Items editor and on the object's Tasks tab. This allows for customization, so if a user wanted to show only human-related tasks, they can leave out mappings for any states, parallels, or clusters that are not human tasks. Additionally, the sequencing of mappings determines the order in which the tasks are displayed in the STEP Workflow tab.



Note that while views and mappings can be applied to a blank workflow canvas, it does not generally make sense to do so. Therefore, it is expected that a workflow has been created prior to any views or mappings being applied. See the **Creating a Workflow** topic for detailed instructions on workflow creation.

Before creating views or mappings, it is important to be aware of the following points:

- The views and mappings configured in the STEP Workflow Designer only apply to workbench and not Web UI. Views in Web UI are defined by the configuration of the screen. If users will not be interacting with

workflows in the workbench, it is not necessary to configure workbench views. Instead, see the **Workflows in Web UI** section of the **Web User Interfaces** documentation.

- A view can contain both object and workflow-specific data. This means that a view can contain attributes, attribute groups, and aspects information specific to the object (such as name, ID, approval status, etc) as well as elements like workflow assignee, deadline, workflow variables, and workflow attachments.
- A view can be mapped to multiple states. For example, if a state is assigned to a particular user or group early on in the workflow to edit data, and again later in the workflow to review data, it is possible that the same data points should be available to them in each state. In this case, the view only needs to be configured once, and that view can then be mapped to each applicable state.
- Once a view or mapping has been started for a workflow, even if they are then deleted to be completed at another time, no states will show in the STEP Workflow tab until views and mapping are completed.

The details for configuring views and mappings are covered in the subsequent topics, **Configuring Views for Workflows** and **Configuring Mappings for Workflows** topics.

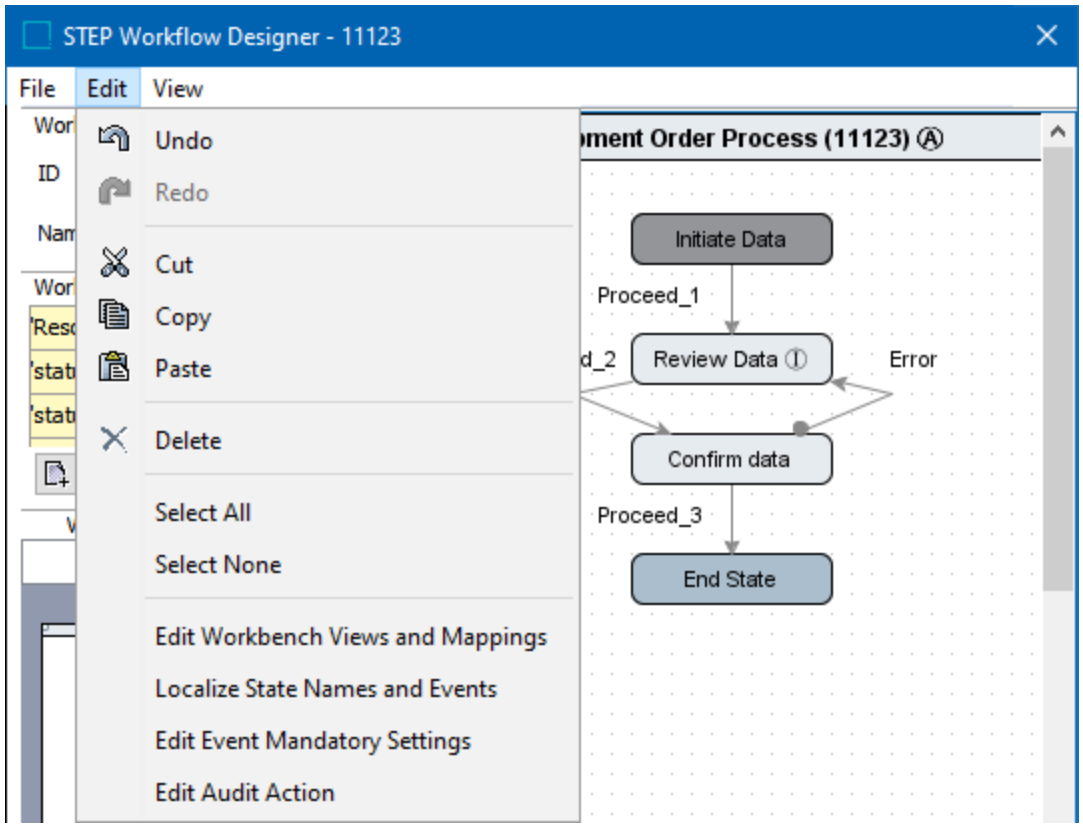
Configuring Views for Workflows

This topic details how to configure views for states. Without views configured, end users will only see the ID of objects in a workflow when accessing tasks via the STEP Workflow navigator tab or the Tasks tab on the object. Therefore, if users will be interacting with the workflow in the STEP Workbench, it is recommended to have views in place so that the user can easily see the data points relevant for their tasks in that state.

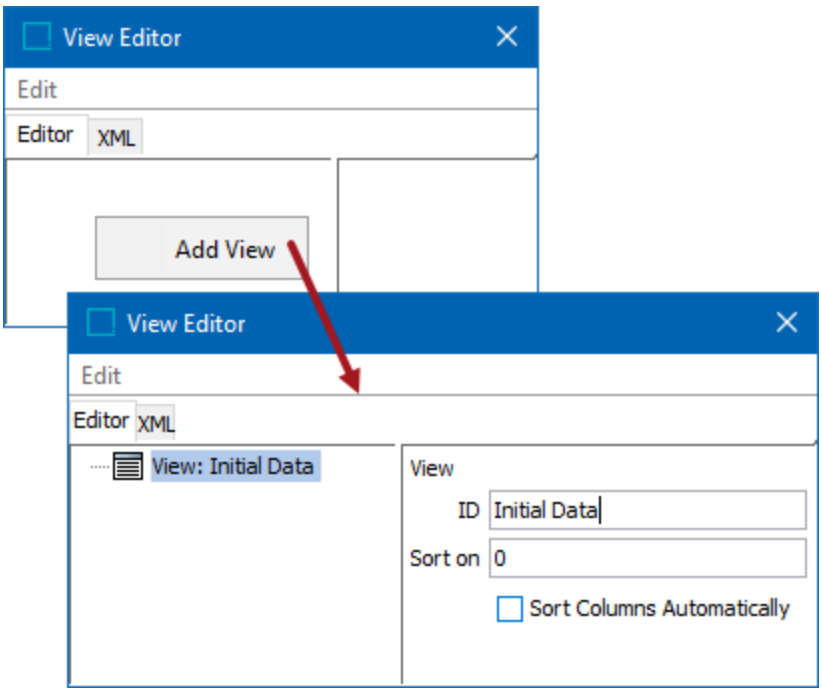
Note: There is no need to configure views for workflows that will be accessed only in Web UI. For information on working with workflows in Web UI, see the **Workflows in Web UI** sectionof the **Web User Interfaces** documentation.

The steps below describe how to create views for a workflow accessed in workbench, and provide reference material for the various view options.

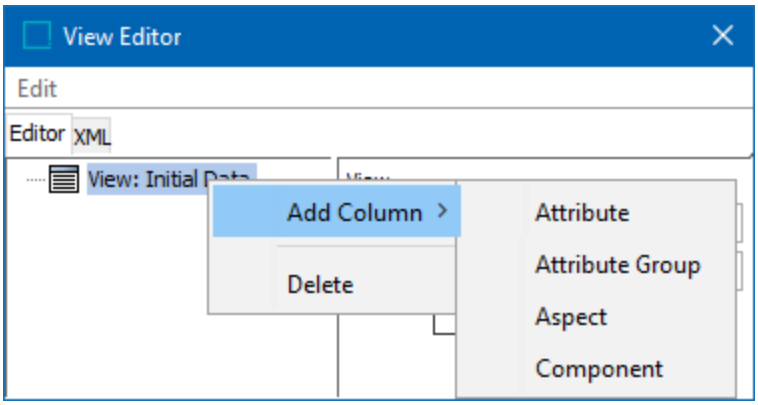
1. To begin creating views, select the desired workflow in System Setup, and open the STEP Workflow Designer by going to **Edit** > **Edit Workbench Views and Mappings**.



2. When the View Editor opens, it will be blank. Right-click on the left side of the panel and select **Add View**. Provide an ID for the view.



3. Once a view has been created, columns can be added by right-clicking on the view, selecting **Add Column**, and choosing a type of data to add to the view. As many columns can be added as needed to provide all necessary information to the user.



The Attribute, Attribute Group, and Aspect options are all for displaying object data, while the Component option is for displaying data specific to the workflow task (e.g., a workflow variable, attachment, assignee, etc).

The below sections describe the various view options in more detail, as well as how to reorder columns within a view.

Important: For views to take effect, each view must be mapped to one or more states. For details on mapping views, see **Configuring Mappings for Workflows**.

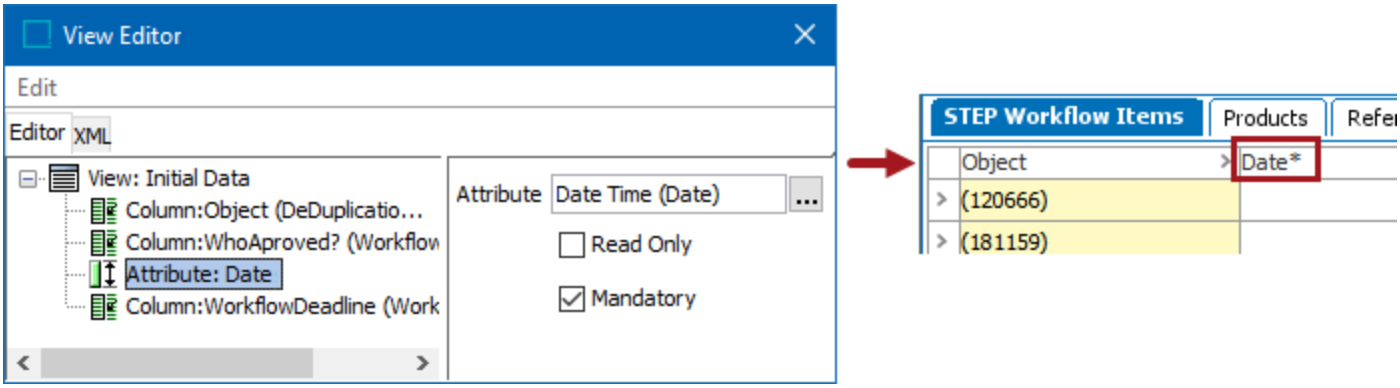
Attribute

The 'Attribute' column type is used to insert a single attribute into the view. The validity of the attribute must match the validity of the objects in the workflow, or else the attribute will not be displayed in the view. Additionally, if working with specification attributes on products, the attribute must also be linked appropriately in the hierarchy to be valid for the workflow objects.

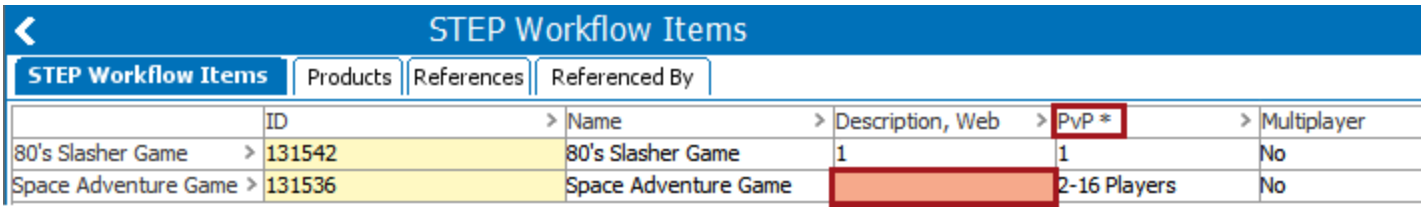
An attribute can be configured as 'Read Only' and/or 'Mandatory'. Attributes marked as 'Read Only' are not editable on the STEP Workflow Items or Tasks tabs. The 'Mandatory' setting will give a visual indication that the attribute is required by showing an asterisk (*).

Note: The Mandatory checkbox in view configuration provides a visual indicator only—it does NOT enforce that the attribute is populated. If the attribute *must* be populated prior to exiting the state, the attribute must be marked as mandatory using the Mandatory Data tab of the state or transition editor. More information on this is available in the **Mandatory Attributes and References in Workflows** topic.

To add an attribute to a view, right-click on the view > **Add Column** > **Attribute**. Click the ellipsis button (...) to select an attribute, and mark the Read Only and/or Mandatory options as needed.



It is worth noting that if a field is highlighted in red, this means that the value is mandatory for approval of the object, independent of the workflow or state. This type of mandatory is set on the attribute by setting Mandatory = Yes. As with the mandatory indicator in the view, this does not force the user to populate a value within the state.



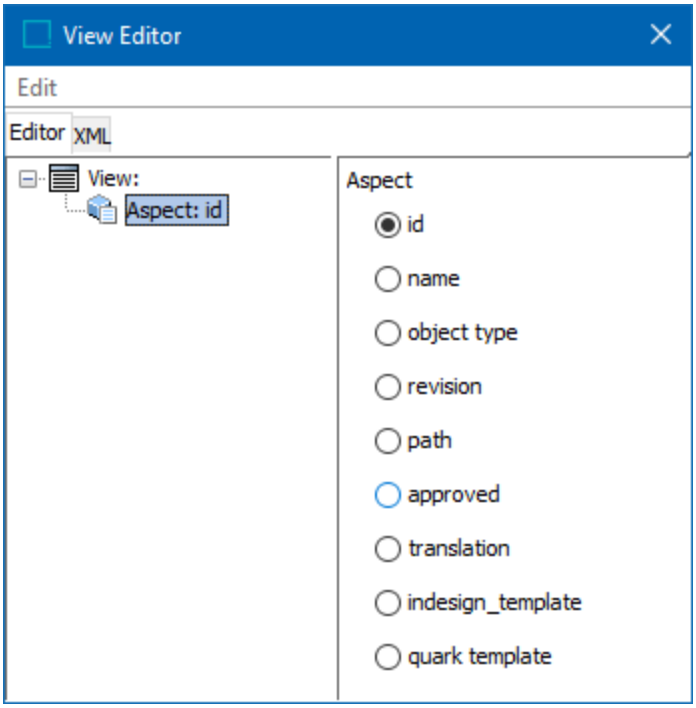
Attribute Group

The 'Attribute Group' column type is used to insert all attributes in a group into the view. The same considerations apply as for a single attribute, described above.

Aspect

Via the Aspect option, it is possible to insert columns for ID, name, object type, revision, and other basic data for the object.

To add an aspect to a view right-click on the view > **Add Column** > **Aspect**. Click the appropriate radio button for the desired aspect. A full description of what each aspect option is can be found in the table below.



Aspect	Description
id	Displays the ID of the object, which is not editable.
name	Displays the STEP Name of the object, which is editable if privileges allow.
object type	Displays the object type of the object. If privileges allow for editing, a dropdown is present from which valid object types can be selected.
revision	Displays the revision number, as well as when the object was last edited and by whom. The field is not editable.
path	Displays the path of where the object exists in STEP. The field is not editable.
approved	Displays the approval status of the object and date / time of last approval if

Aspect	Description
	applicable. The field is not editable.
translation	Displays the translation status of the object, for example: Up to date, In Progress, Not Translated, etc. (not editable). For more on translations, see the **Translations** documentation.
indesign_ template	Displays the STEP Publisher product template being used for print. By clicking on the hyperlink, the user is brought to the template.
quark template	Legacy option that should not be used in current implementations.

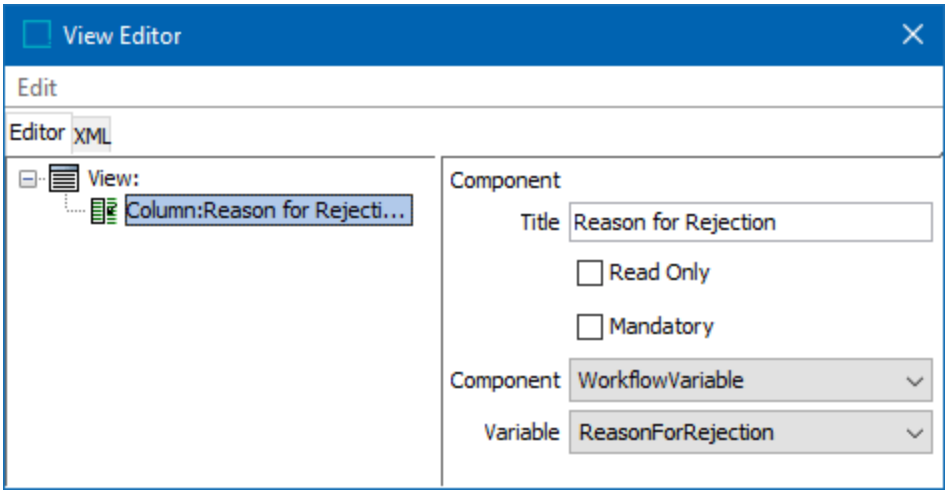
Note that Unique Key is not a selection option. If it is required to display the key value of an object, the attribute(s) driving the key should be selected using the Attribute or Attribute Group options.

Component

The Component option gives access to a list of data specifically related to workflows, including assignee, workflow attachments, workflow variables, etc.

To add a component to a view right-click on the view > **Add Column** > **Component**.

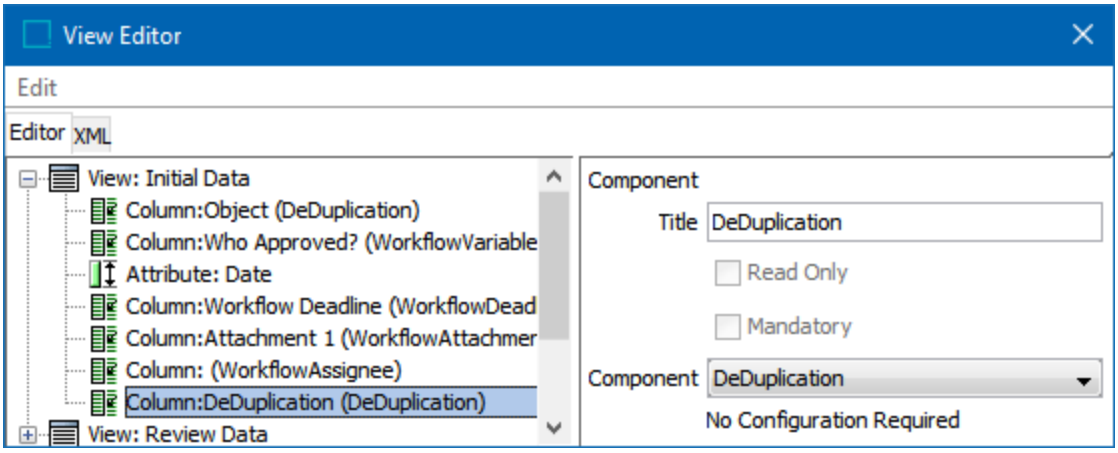
Populate the **Title** (header for the column in the view), select **Read Only** or **Mandatory** as appropriate, and select a component from the dropdown. Note that selection of a component may result in additional selection options. For example, selection of Workflow Variable yields an additional dropdown to select the specific variable to be used. Each component option is described in detail below.



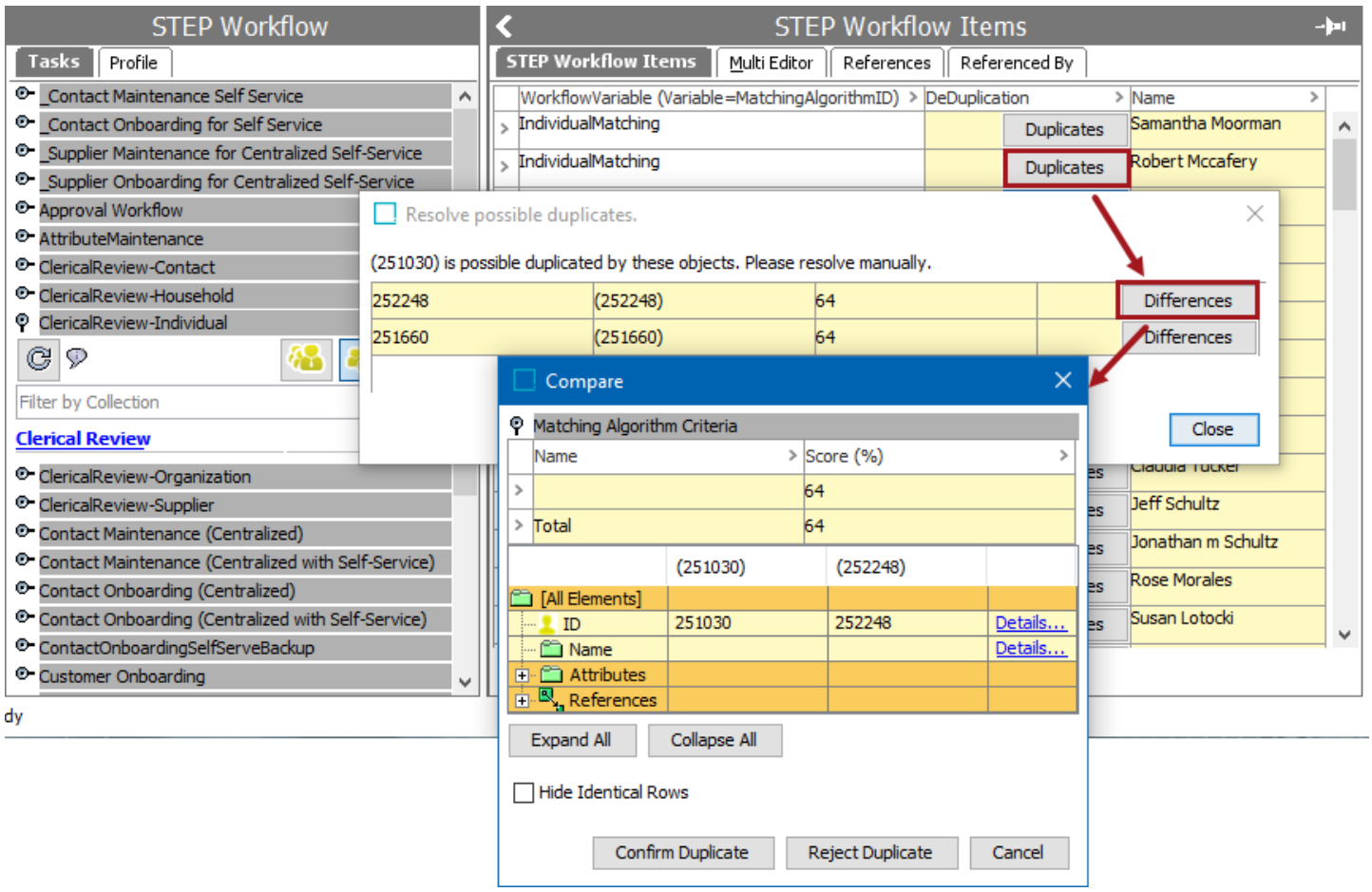
Read Only and **Mandatory** fields are not applicable on all components and are disabled in that case. When available, fields marked as 'Read Only' are not editable on the STEP Workflow Items or Tasks tabs. Checking 'Mandatory' means that an asterisk (*) is shown with the Title to indicate to users that the field should be populated. However, it does *not* enforce that the field is populated prior to the task exiting the state.

DeDuplication

Adding the DeDuplication component to a workflow view allows a user to see if there are any objects in the system that have been flagged as potential duplicates by a matching algorithm.



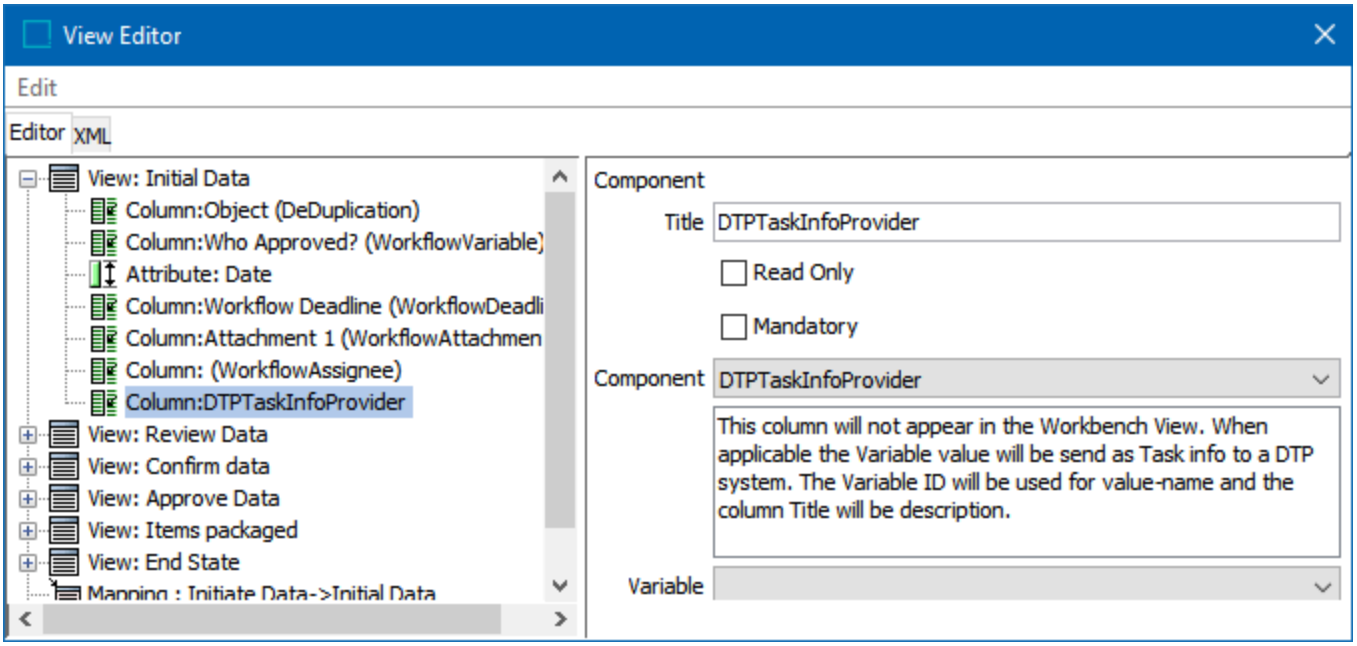
If potential duplicates have been identified, a Duplicates button is present that the user can click to access a list of possible duplicates for the selected object in the workflow. Clicking the Differences button opens a Compare screen where the user can view differences and confirm or reject duplicates.



For additional information on deduplication workflows, see the **Matching, Linking, and Merging** documentation.

DTP Task Info Provider

Component that can be used if a user wishes to show task data in the InDesign plugin tasks tab. It will not be rendered in the workbench. The Variable ID will be used for value-name and the column Title will be description.

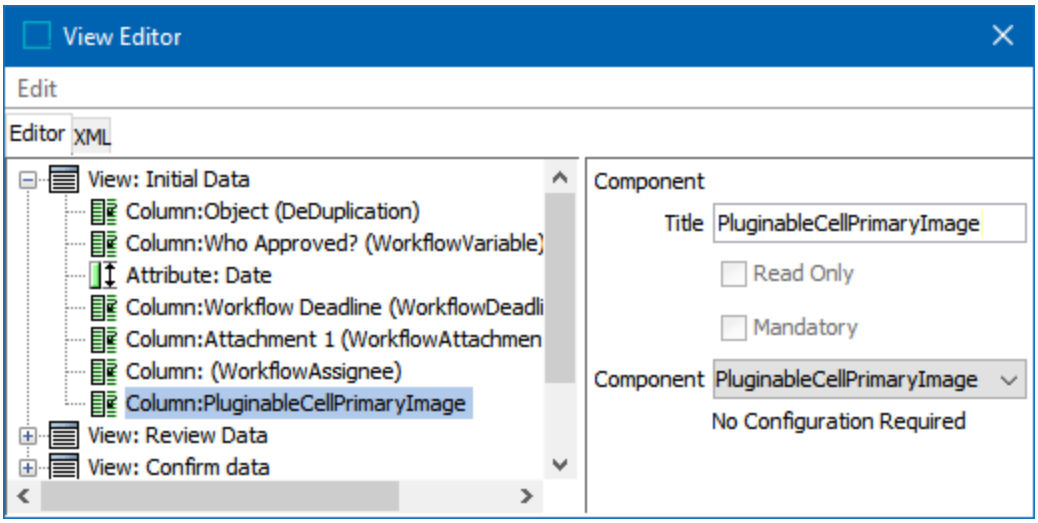


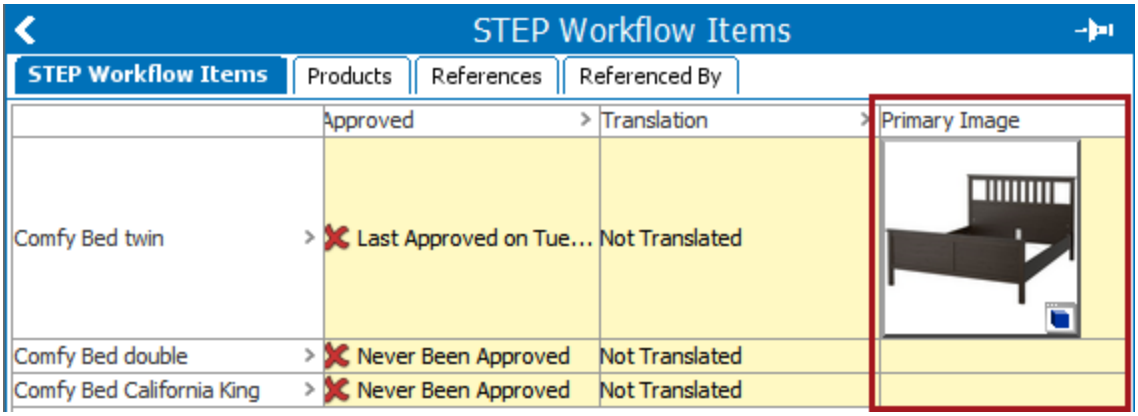
Pluginable Cell Debug

Experimental debug option that generally should not be used.

Pluginable Cell Primary Image

Component for displaying the primary image of the object.

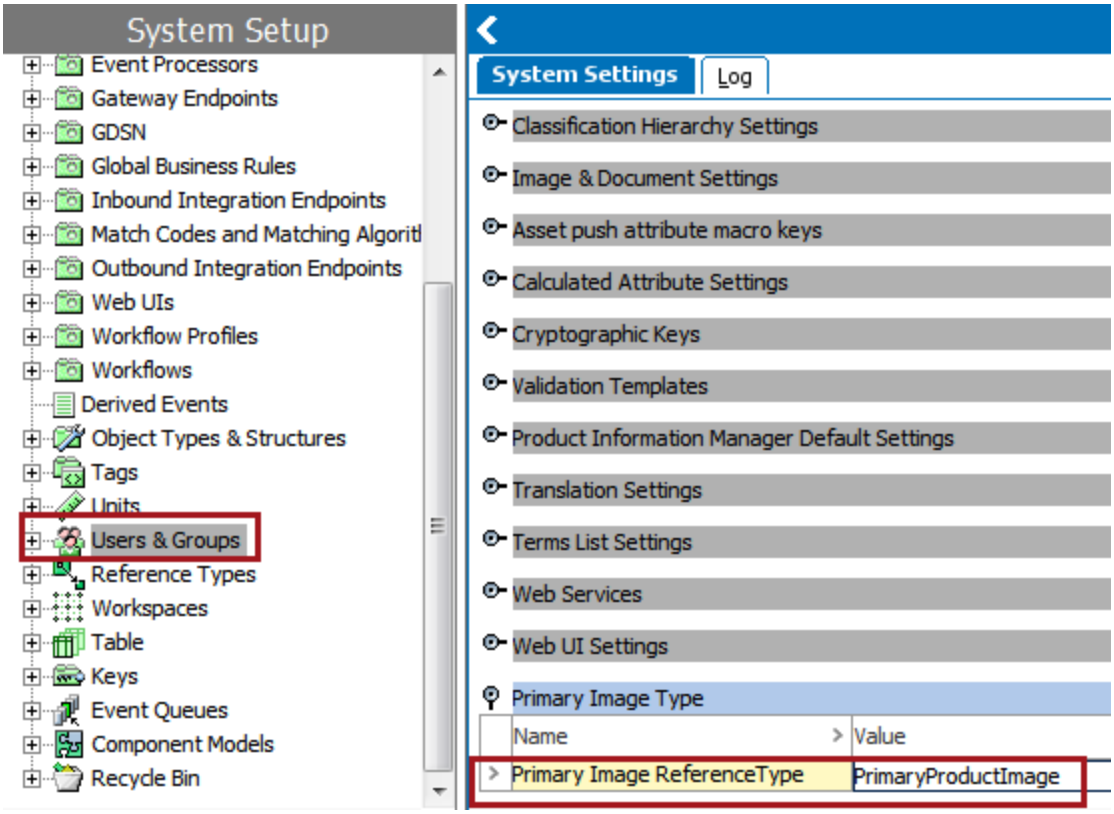




The component displays a thumbnail of the primary image. Note that the image and reference are both not editable (e.g., images cannot be added or removed using this component).

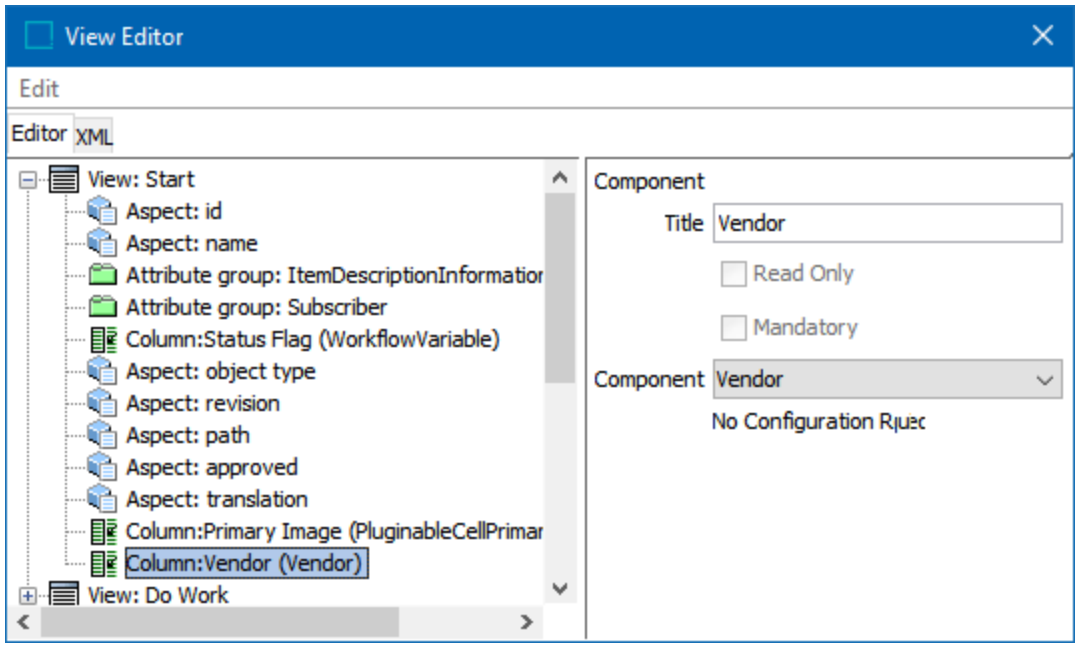
Note: If end users need to edit references (primary image or otherwise), this can be done using the standard editor functionality on the References and Referenced By tabs.

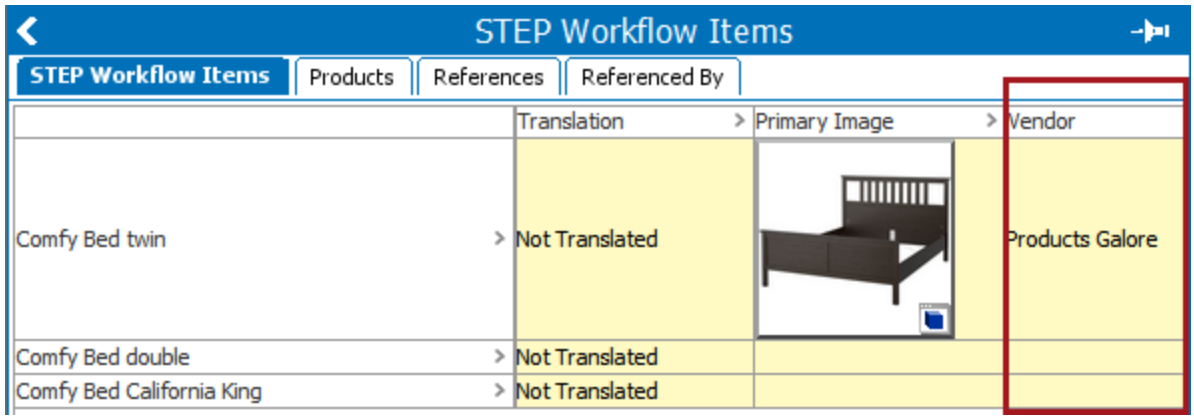
The reference type used to identify the primary image is set in System Settings (System Setup > Users & Groups > Primary Image Type). The STEP ID of the reference type that should be used must be entered in the Value field.



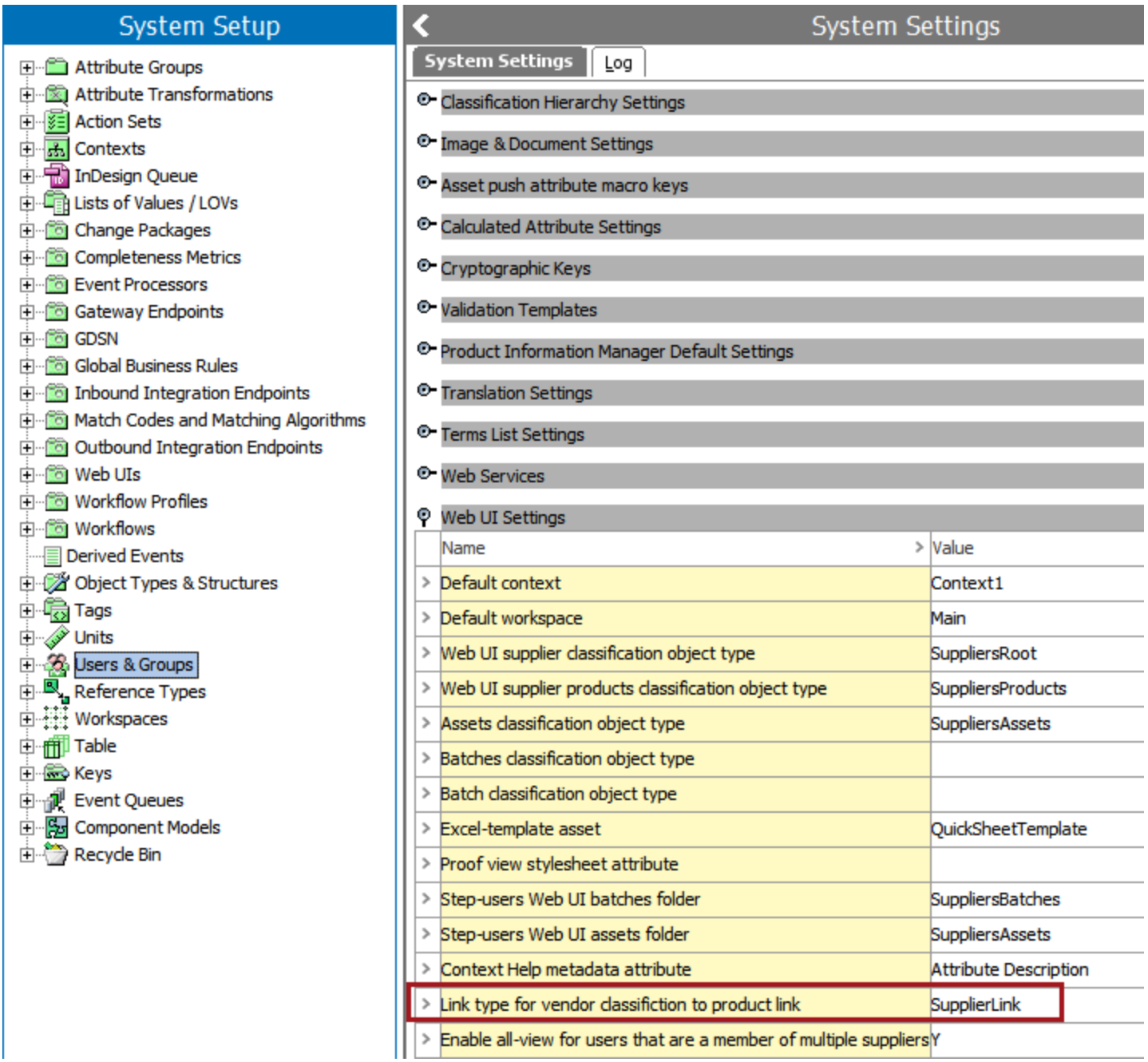
Vendor

Vendor is used to display the name of the vendor / supplier that the object is assigned to. This is typically used for systems that have a Supplier hierarchy in place. More information on this setup is available in the **Supplier Web UI Setup Guide** in the **Web User Interfaces / Web UI Getting Started** documentation.





The component displays the name of the supplier classification folder that the object is linked to (object will be in the Products sub-folder), using the reference type indicated in the 'Link type for vendor classification to product link' system setting.



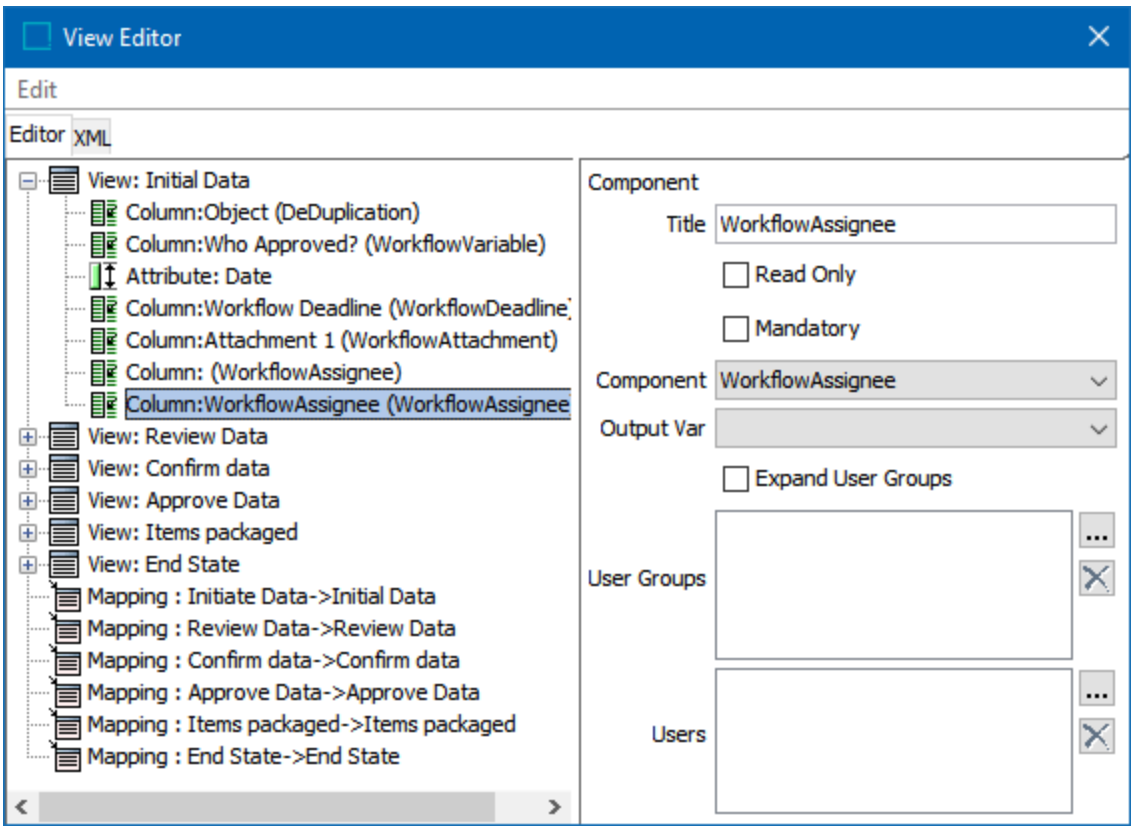
Workflow Assignee

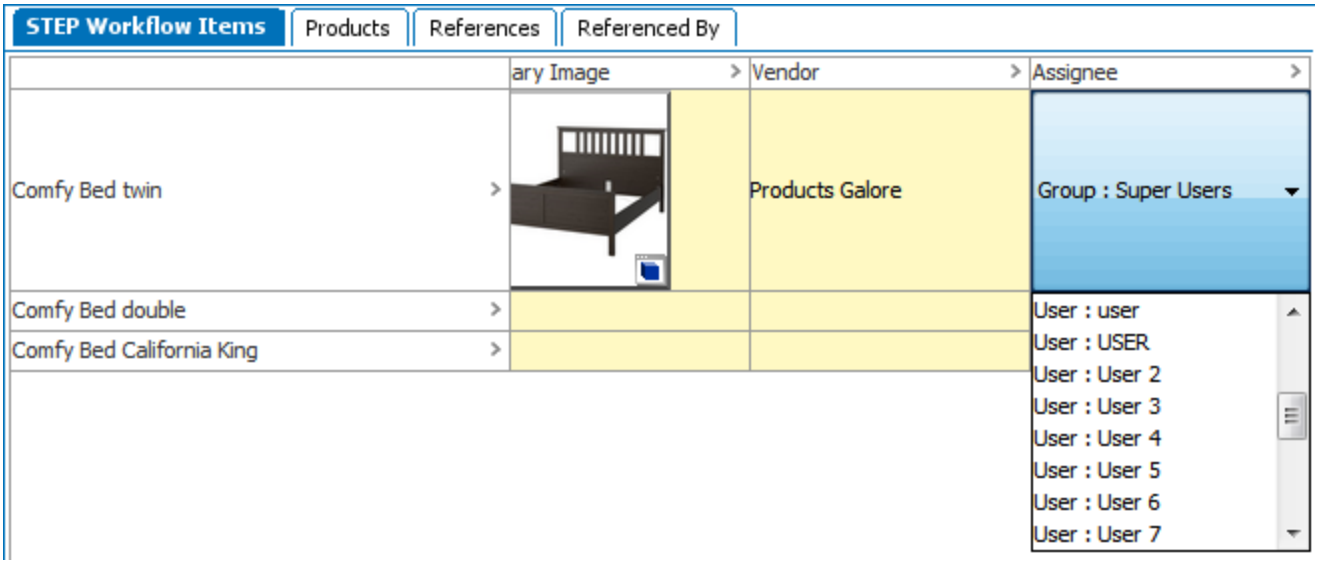
Used for displaying the current assignee and/or letting users with the STEP Workflow Administrator privilege re-assign the current task to another user / group.

If to be used for re-assigning, either the **User Groups** or **Users** parameter (or both) must be populated. Selections will be the available assignees.

If the **Expand User Groups** checkbox is selected, all users in the selected user groups will be available for selection / assignment (as well as the user group itself). Note that when a privileged user selects a user / group to re-assign the task to (in the STEP Workflow Items editor or on the object Tasks tab), the re-assignment will occur immediately, and the task might disappear upon refresh.

The **Output Var** parameter can be used to select a workflow variable to write the public ID of an assignee. If a selection is made, when an end user populates the cell, no reassignment takes place and no special privileges are required. The use for this is that a user is able to select a 'future' assignee - the actual assignment must be done later by a business action that looks up the assignee using GroupHome / UserHome. If the same ID is used on both a group and a user, the component will use the group, as it's found first.

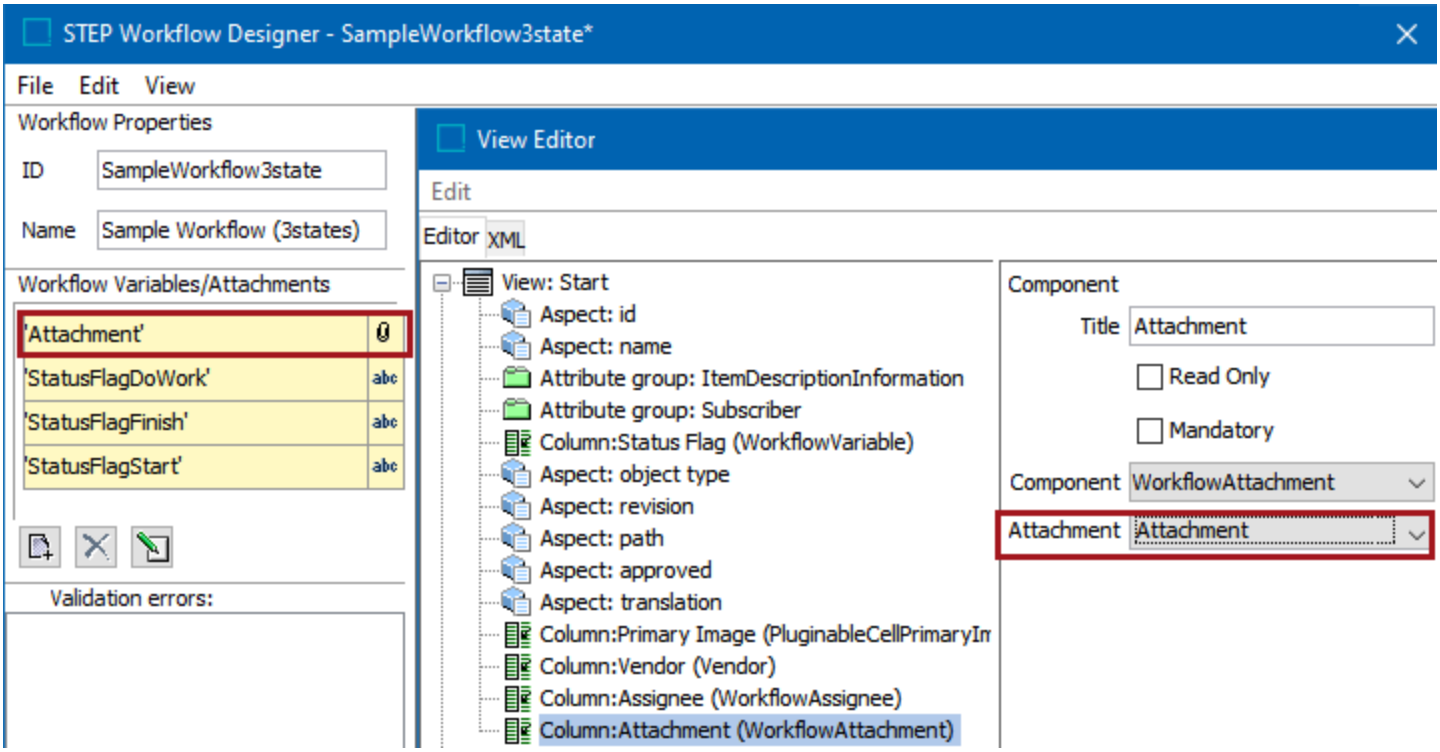


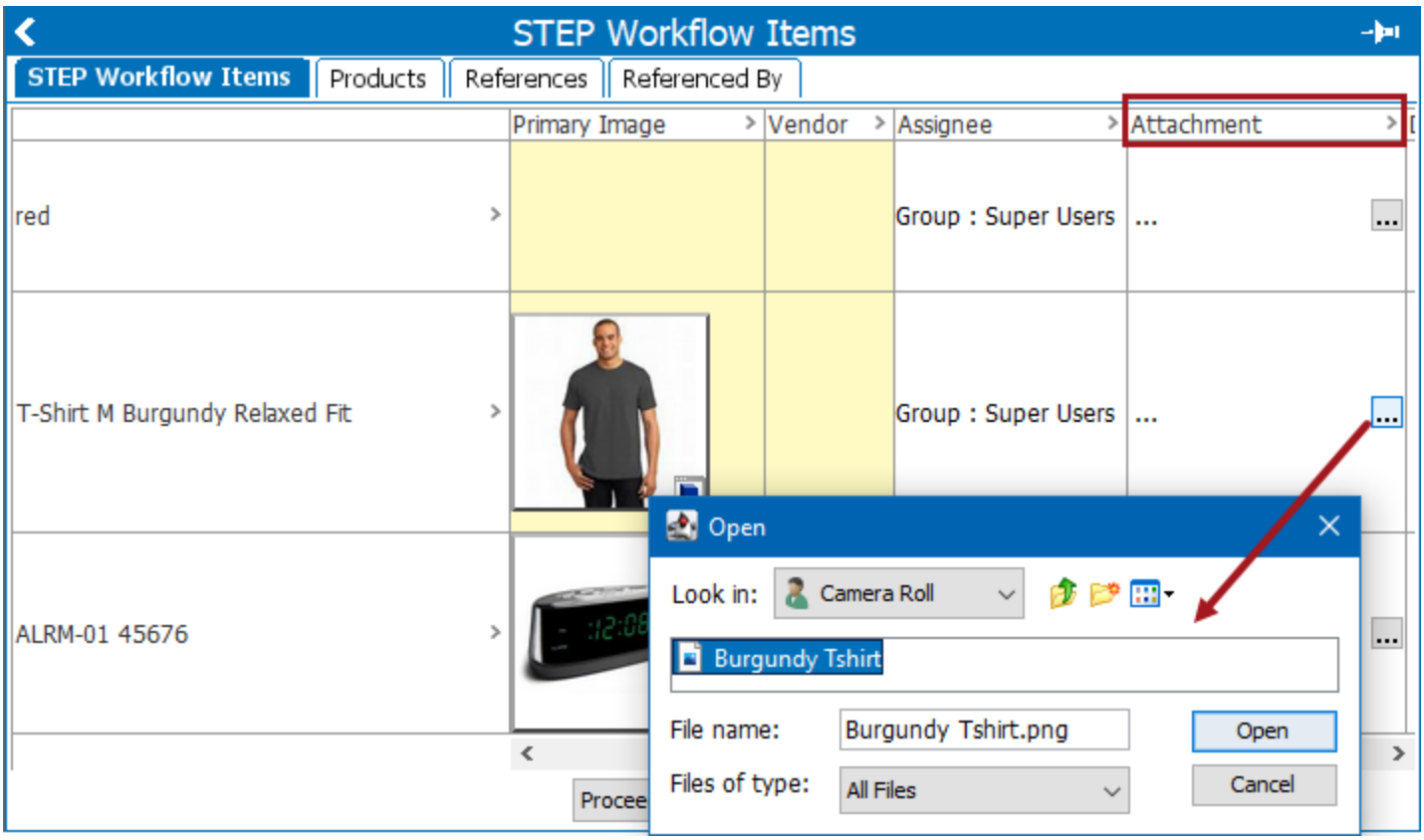


For more on Assignees and the description of the various options for the Workflow Assignee component, see the **Assignees in Workflows** topic in the **Workflows** documentation.

Workflow Attachment

A component that lets users add, open, download, and delete Workflow Attachments. Use of this requires that a Workflow Attachment has been previously set up in the STEP Workflow Designer (and saved) so that it can be selected in the **Attachment** dropdown.





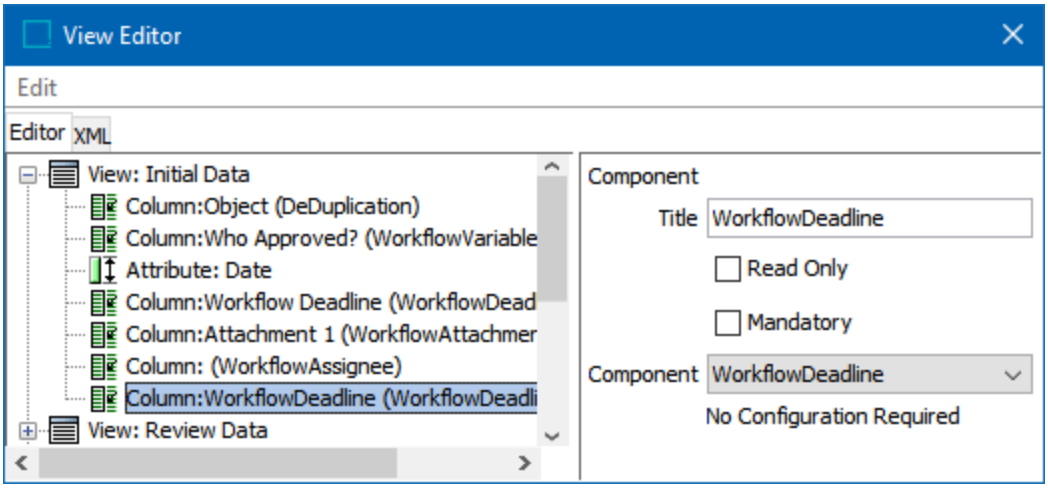
Note: If marked as 'Read Only', the add and delete options will be disabled, but users will still be able to open and download attachments.

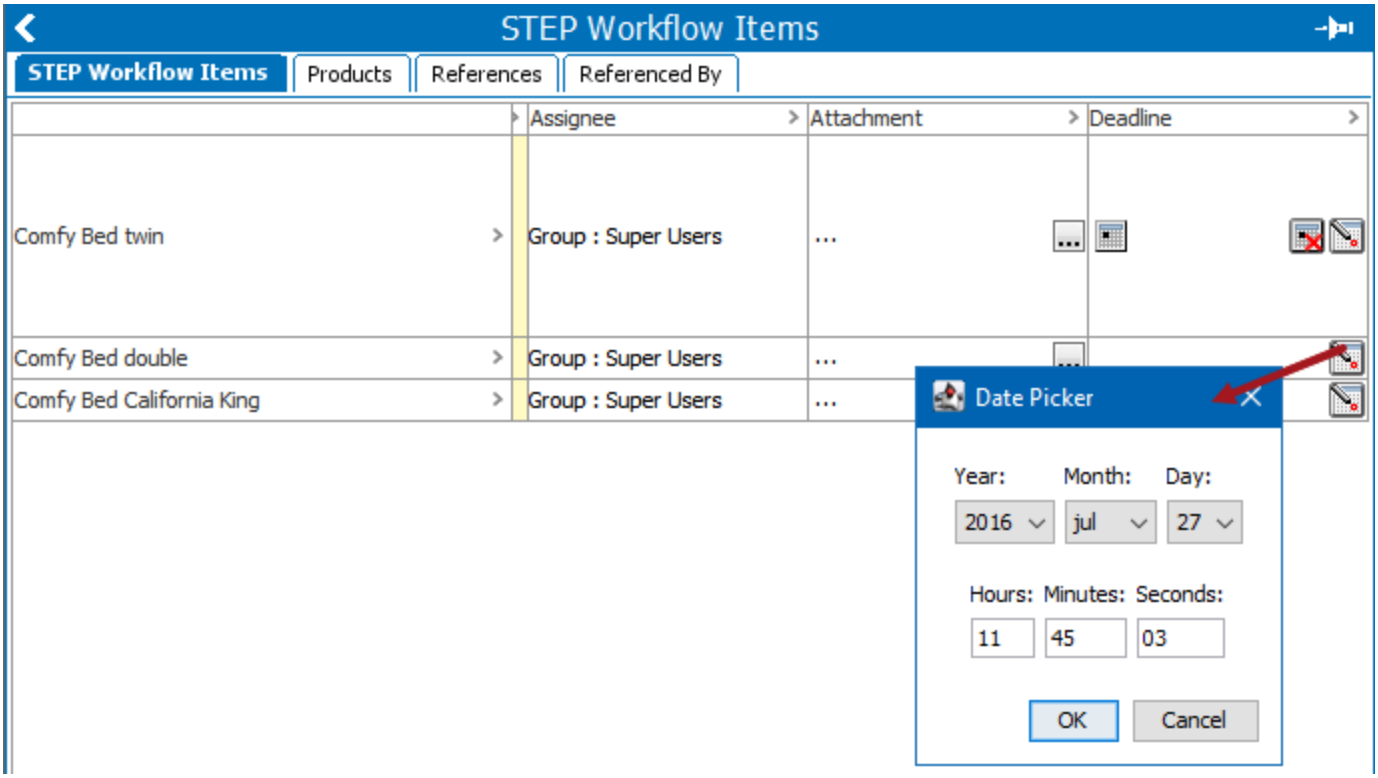
For more information on Workflow Attachments and the features in this component, see **Workflow Attachments** in the **Workflows** documentation.

Workflow Deadline

Users with the STEP Workflow Administrator privilege will be able to set, edit, and delete task deadlines. Users without the privilege will see different icons depending on whether the deadline is reached or not. Deadlines will be displayed when a user hovers over the icons.

Note that the editability of the deadline field is determined solely by privileges, and therefore the Read Only setting is not applicable for this component.

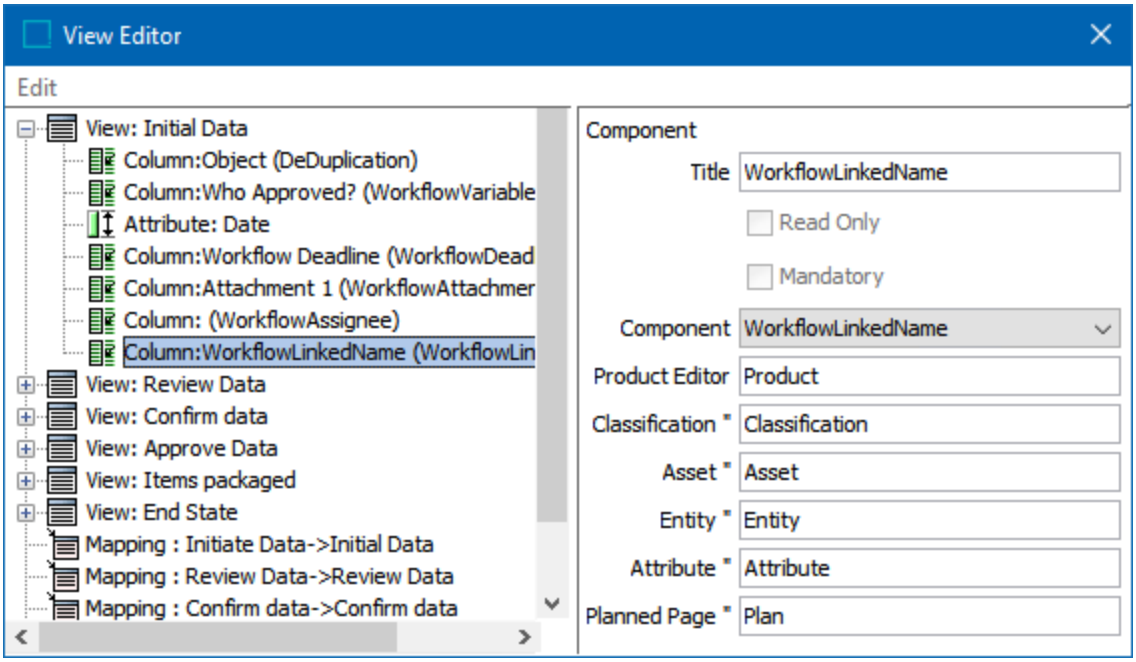




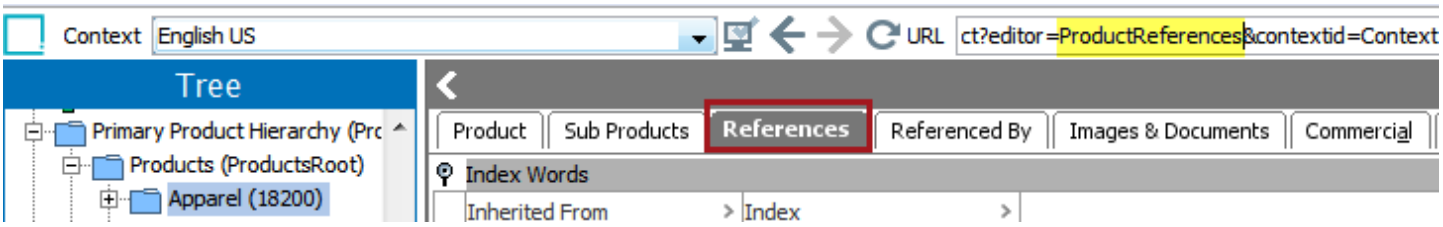
For more on this see **Deadlines and Escalations in Workflows** in the **Workflows** documentation.

Workflow Linked Name

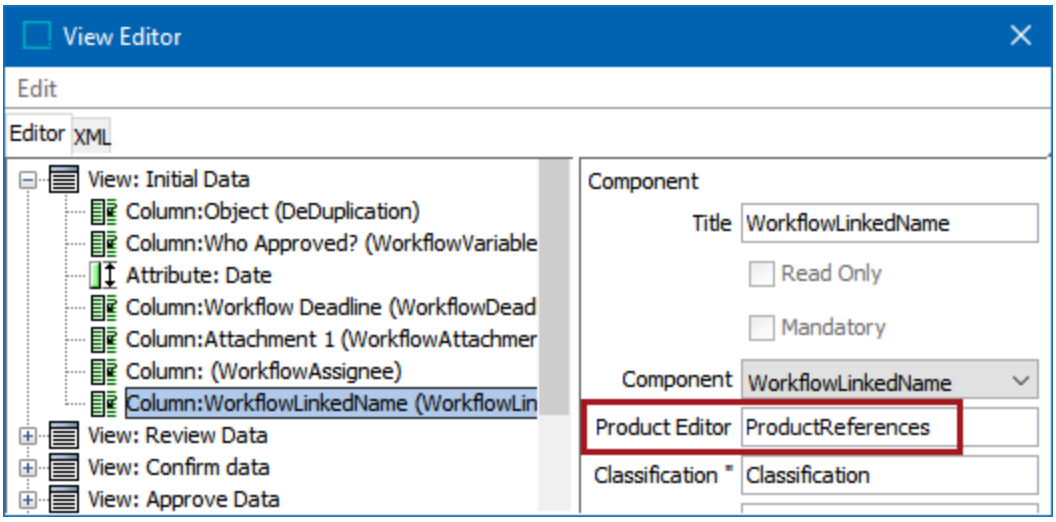
Component for displaying the object name as a link to the configured object editor. When the component is selected, default values are populated, which takes the user to the standard editor for each super type of objects (e.g., the Product tab in Tree view for "blue" objects in STEP). However, these default values can be edited so that the user is directed to another tab when clicking the hyperlink in the view (e.g., to the References tab or Images & Documents).



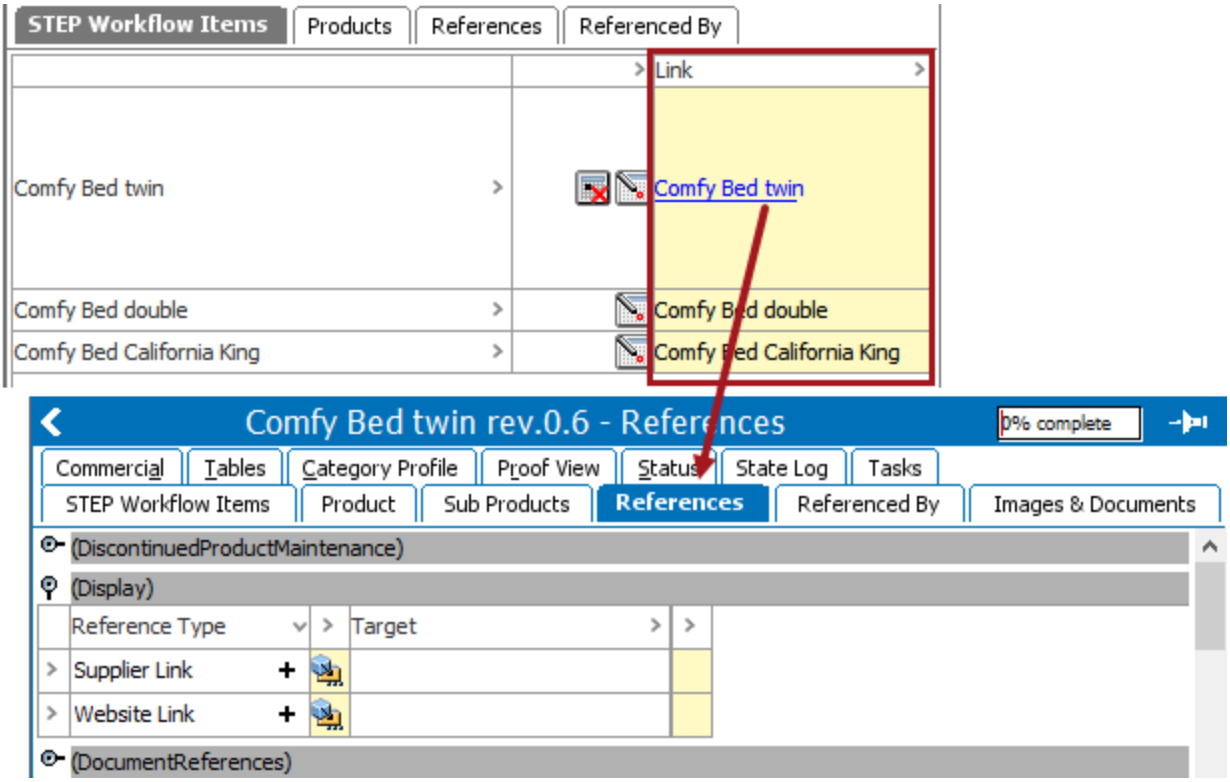
For example, to change the default landing page to be the References tab of the product editor, go to the desired editor, choose the desired tab, and look at the STEP URL field. In this example, the tab is named 'ProductReferences'.



Type 'ProductReferences' for the Product Editor parameter and save the changes to the workflow.

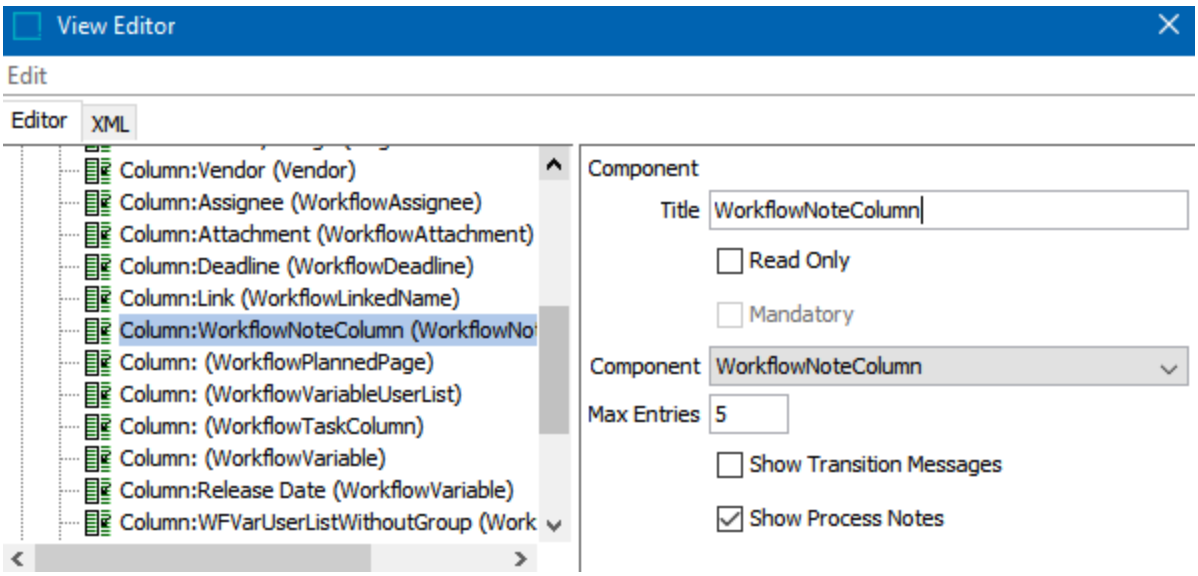


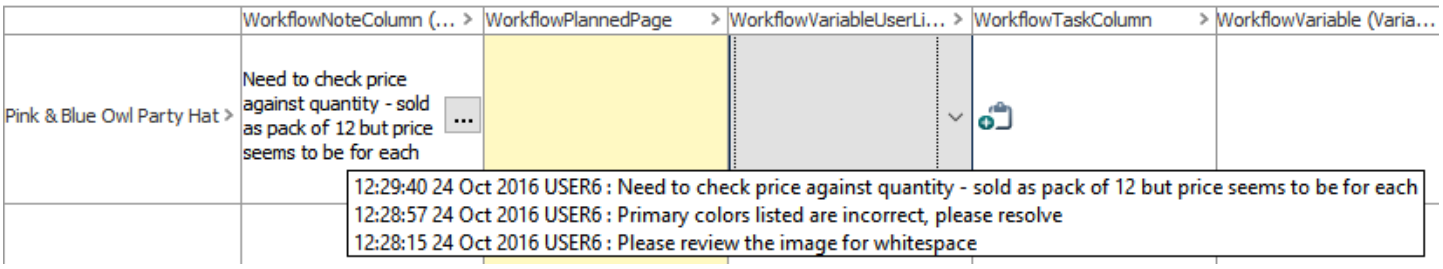
Display the STEP Workflow Items tab and click the name link to display the configured tab within the STEP Workflows editor.



Workflow Note Column

Component for displaying the most recent Process Note or Transition Message. If *not* configured as Read Only, a new note can be added, and will persist throughout the workflow. If the field is hovered over in the end user view, the configured number of recent entries entered in the **Max Entries** field will display. Along with the message, the user info and time-stamp will also show.





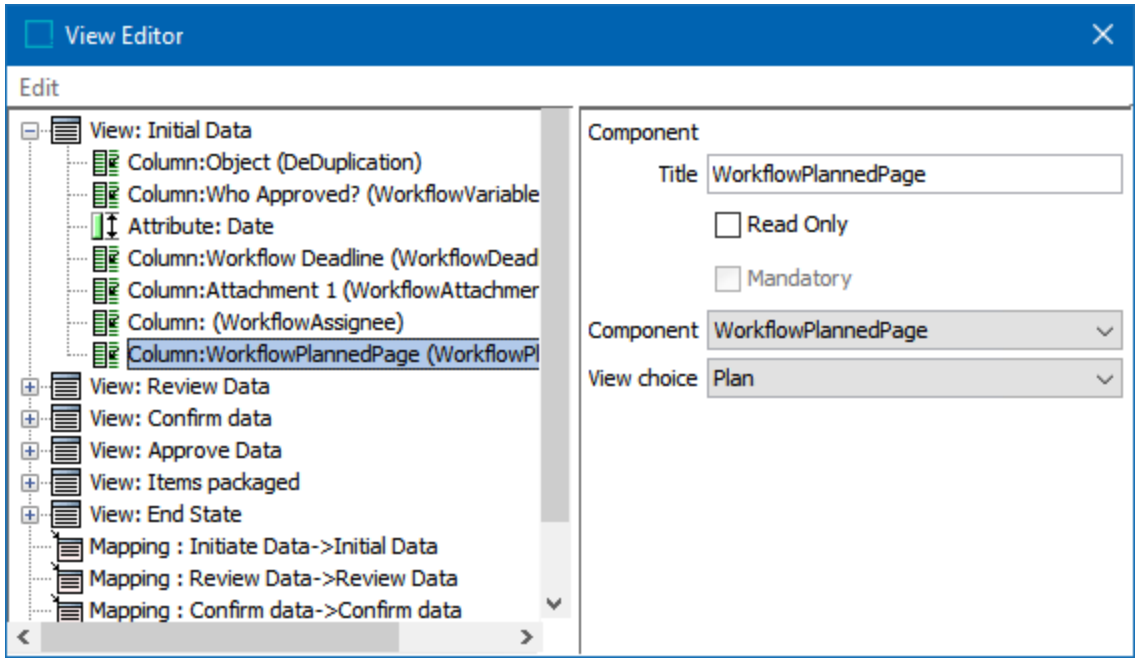
The **Show Transition Messages** and **Show Process Notes** parameters are used to define the types of notes displayed. Care must be taken when making these selections as they significantly impact the user experience. Typically, only one option will be selected, as appropriate for the intended use.

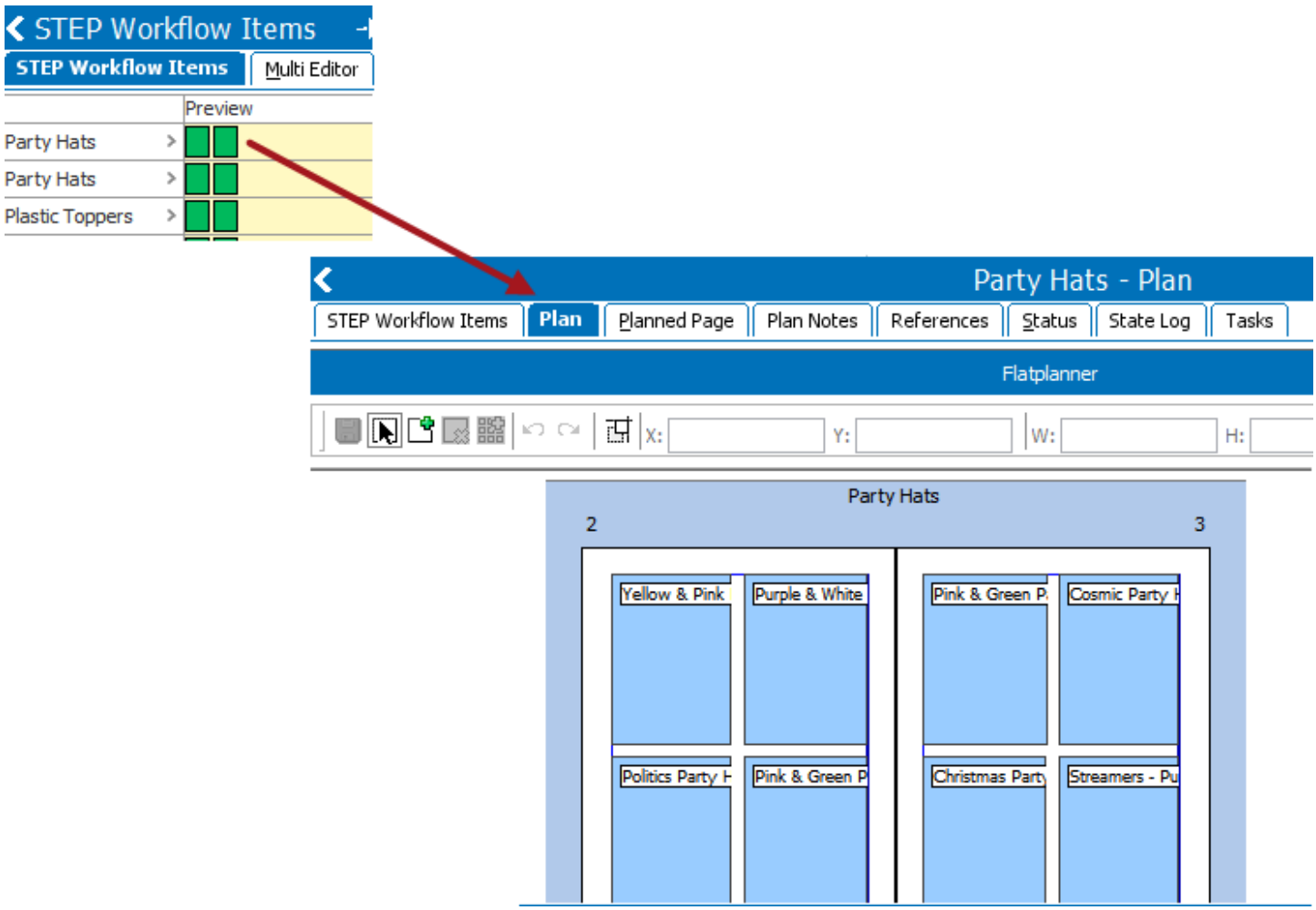
- **Show Transition Messages:** Check this if a message is provided at the time the task is submitted, and the next user should see that message. If no message is provided at the time of transition, a transition is still logged, though without a message (e.g., time, date, and user can be seen upon hover but no corresponding message is displayed). The intent of this is for serial workstreams where one user will need to see a message entered from the immediately preceding user, but is unlikely to need to see previously entered messages from those further upstream. Therefore only the message entered by the last user is displayed in the view.
- **Show Process Notes:** Check this if users will enter notes while working with tasks, and the notes should be visible to the next user. The intent of this is that the current user needs to pass information to one or more subsequent users, and they enter the message while working with the task (e.g., prior to submission). The most recently entered note will always be available in the direct view, with additional notes available upon hover.
- **Both:** Use caution when selecting both parameters in the same column. The most recent transition message is always displayed in the view (if no message was entered, the view displays no value). Any notes entered outside of the submit message are only available upon hover (even if a transition message was not provided). If a transition message is *always* provided, this can be a valid setup. However, if transition messages are not guaranteed, it can appear to the end user that no messages are present, when in fact

process notes have been provided. If both process notes and transition messages need to be available, it is recommended to add the column to the view twice with only one parameter checked in each so that both process notes and transition messages are shown independently.

Workflow Planned Page

Component that can be used in workflows valid for Planned Pages. It will show a thumbnail of the plan, Sticker book, or Mounted pages (as directed by the **View choice** parameter), and links directly to the selection when clicked.

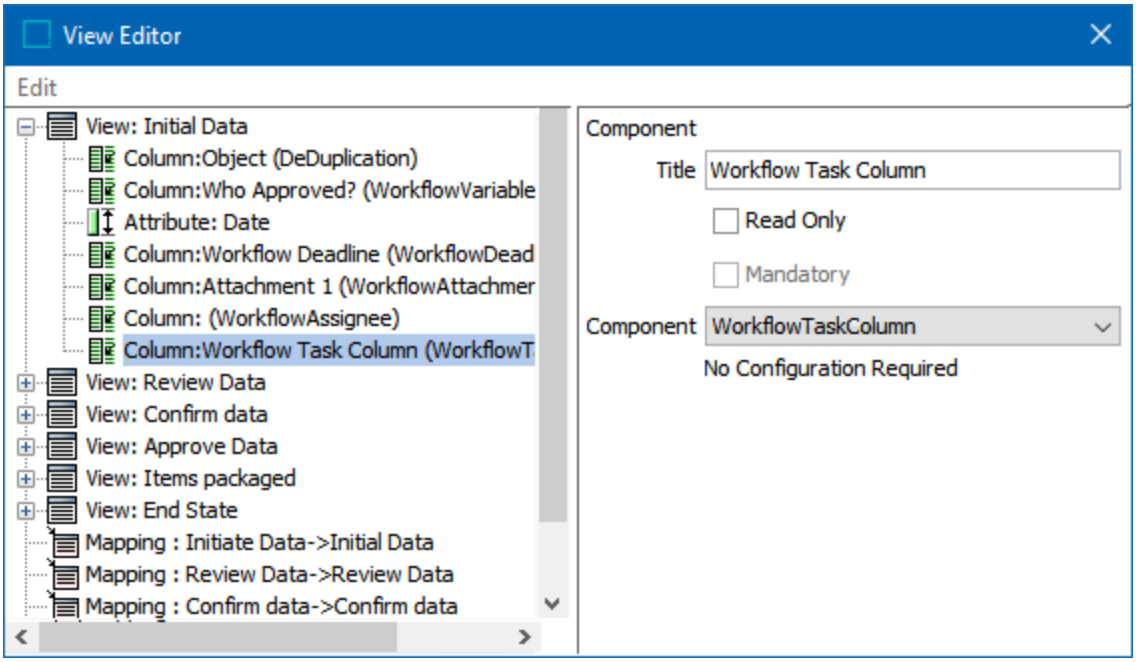


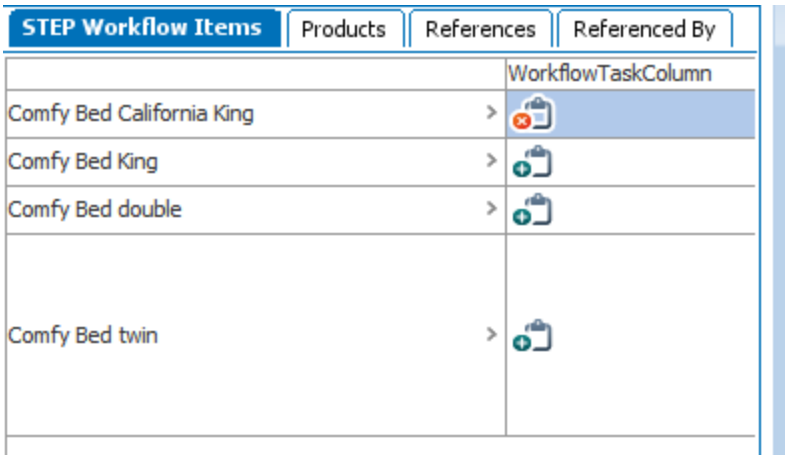


Workflow Task Column

Component for letting users claim / release a task. Users will not be able to work on tasks not assigned directly to them, so if a task is assigned to a user group they are a member of, they have to claim it in order to work with it. When a task is claimed, it is possible to release it again, assigning it back to the latest group assignee. If the task has never been assigned to a user group, the task is assigned to the default assignee for the state.

This component is especially useful for states where the assignee initially is a user group and the desire is to let users in that group claim and complete tasks.

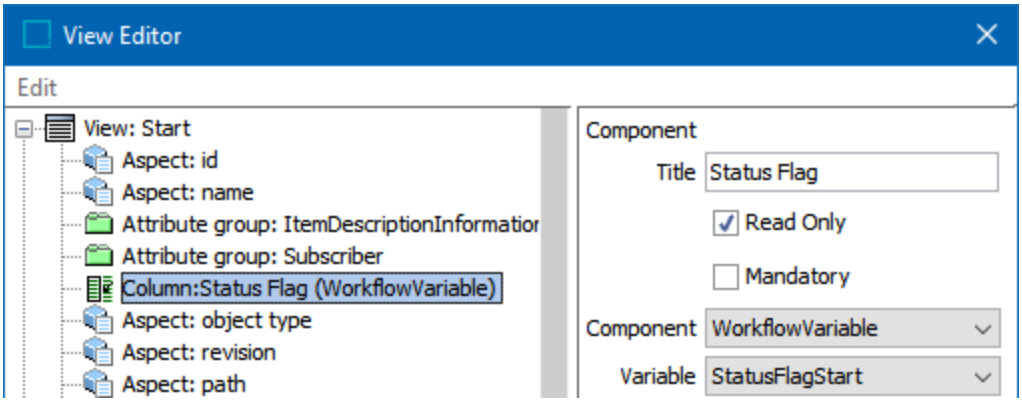




For more on task ownership, see the **Assignees in Workflows** topic in the **Workflows** documentation.

Workflow Variable

Component for viewing and editing Workflow Variables, which requires selection of the **Variable** that should me made available. If the selected variable is bound to an attribute and the component is configured to *not* be Read Only, the cell editor matching the validation base type of the attribute will be used (e.g., a date picker for attributes with a type=Date).



For more on variables, see the **Workflow Variables** topic in the **Workflows** documentation.

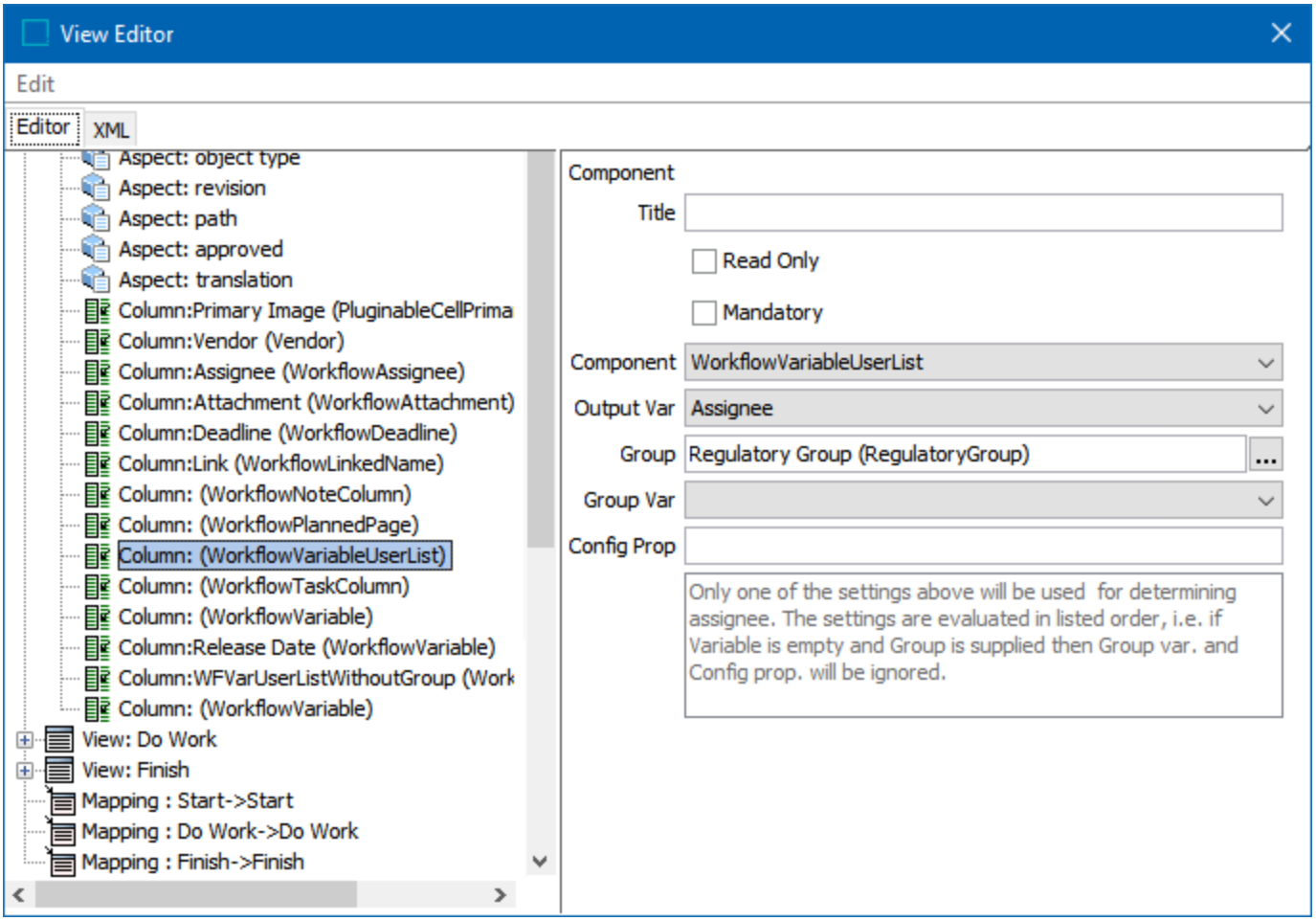
Workflow Variable User List

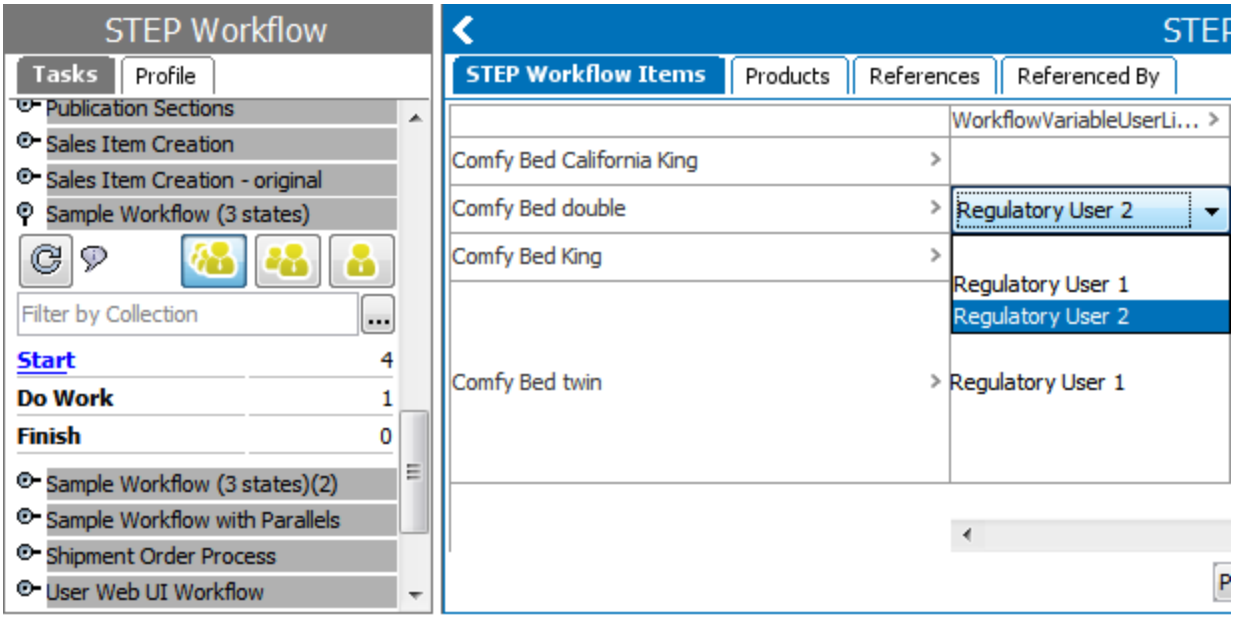
Component for letting the user select from a list of users. The typical use case for this component is one where a user, as part of one task, selects who should handle a subsequent task. The ID of the selected user is then stored in the workflow variable specified in the **Output Var** parameter. From there, it can be used to apply an assignee in a subsequent state using the **Fetch from Variable** assignee option in the State Editor Assignee tab.

An existing workflow variable must be selected in the **Output Var** parameter. This is used to store the selection so that it can be accessed later in the workflow (e.g., at the time of task assignment in a subsequent state).

One of the three selection parameters must be configured (Group, Group Var, Config Prop). The three properties are evaluated in the order shown. Therefore, if Group is populated, it will be used, and Group Var and Config Prop will not be evaluated.

An existing User Group may be selected in the **Group** parameter, and the users in that group will then be available for selection by the end user. This is used when users will always choose a user from within the selected group. If the user group that users need to select an assignee from can vary based on other factors, then the **Group Var** parameter should be used. From this, an existing workflow variable must be selected, and it is expected that at some point in the workflow (prior to the state for which the view is being configured), the variable has been populated with the ID of a User Group. The system then reads the ID from the workflow variable on the task and makes the users from the corresponding user group available for selection. A **Config Prop** option is also available where the ID of a configuration property can be supplied and the user group will be read from there. This is typically only used in conjunction with extensions.





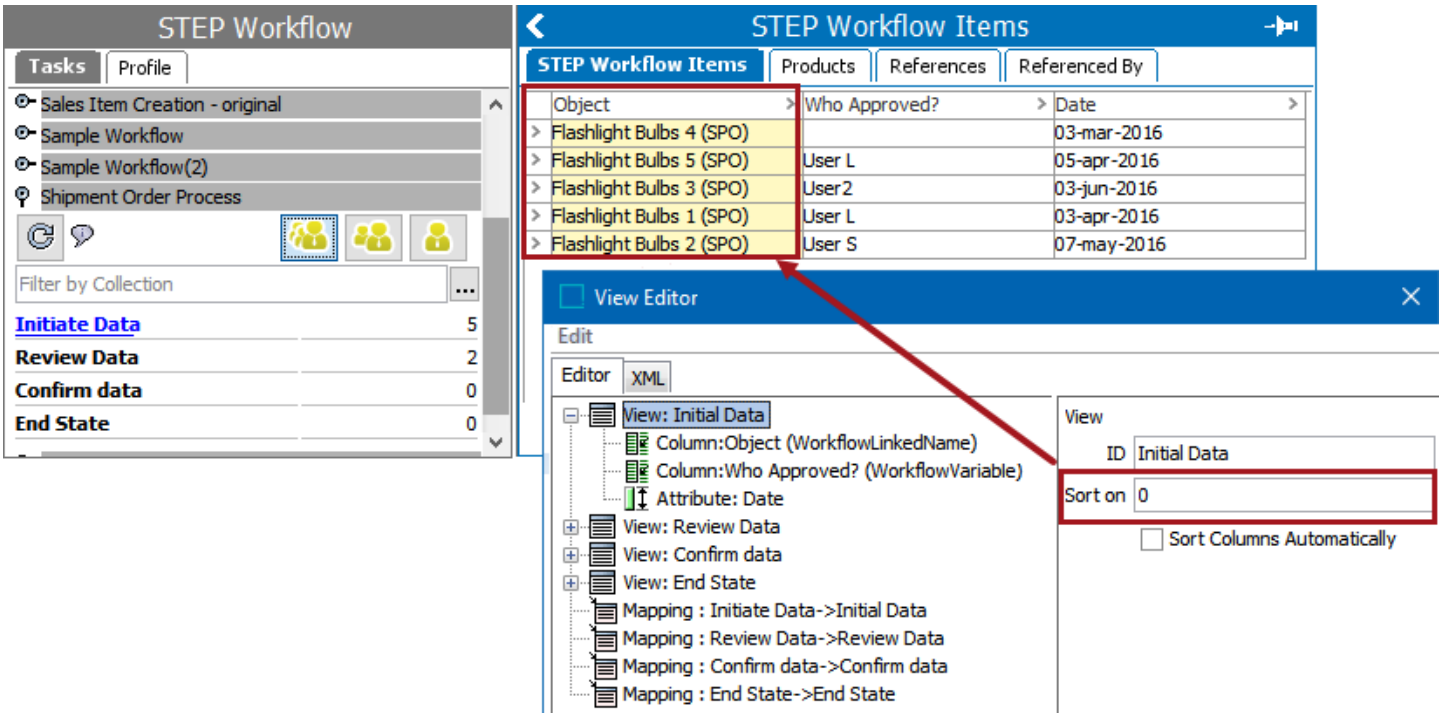
For more on variables, see the **Workflow Variables** topic in the **Workflows** documentation.

For more on Assignees, see the **Assignees in Workflows** topic in the **Workflows** documentation.

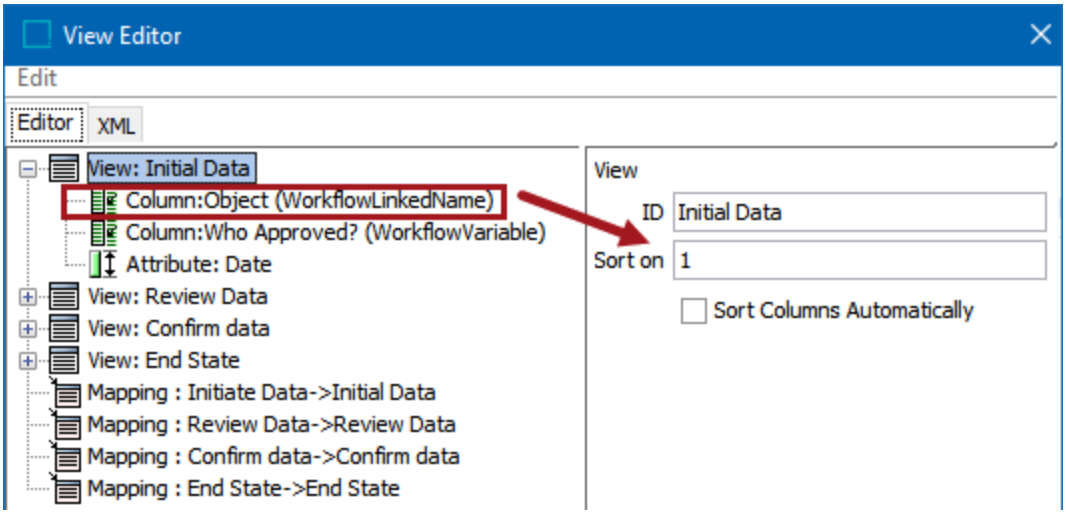
View 'Sort On' Option

The Sort On option allows tasks in the workflow to have a default sort order specified by a particular column. An integer can be entered to indicate the column that should be used for sorting (1= first column, 2= second column, etc.). An example is below.

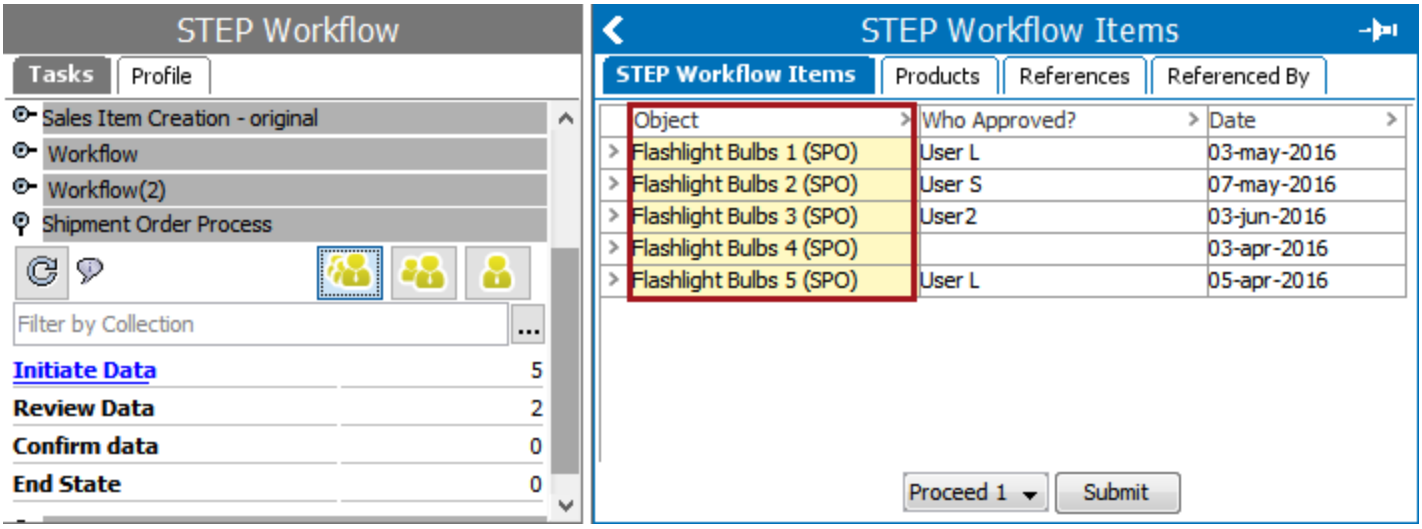
With no sort order specified, the items are in no particular order in the STEP Workflow Items tab.



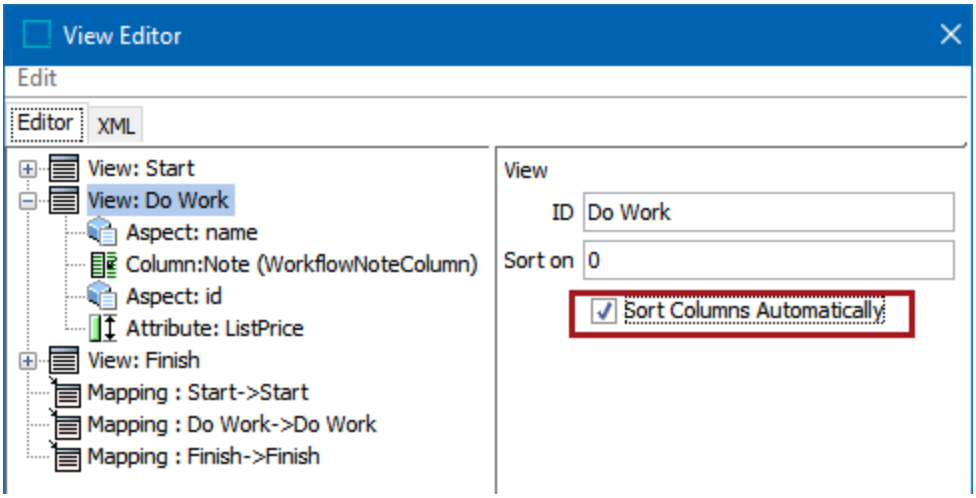
However, when a number is put into the Sort On option to distinguish what column the view should be sorted by, the items are sorted in ascending order. Notice that in the View Editor, in the first view the object name appears in the first column. To sort by this, the user types in 1 to represent the first column in this view.

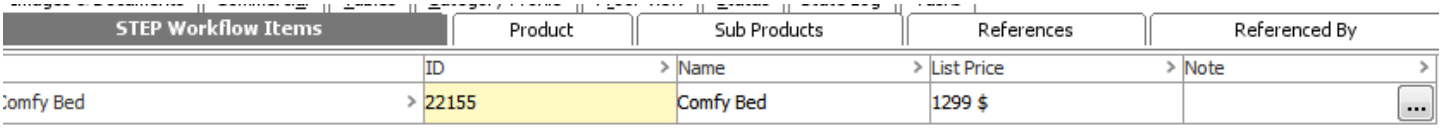


All objects in the workflow in this particular view are now sorted according to object name.



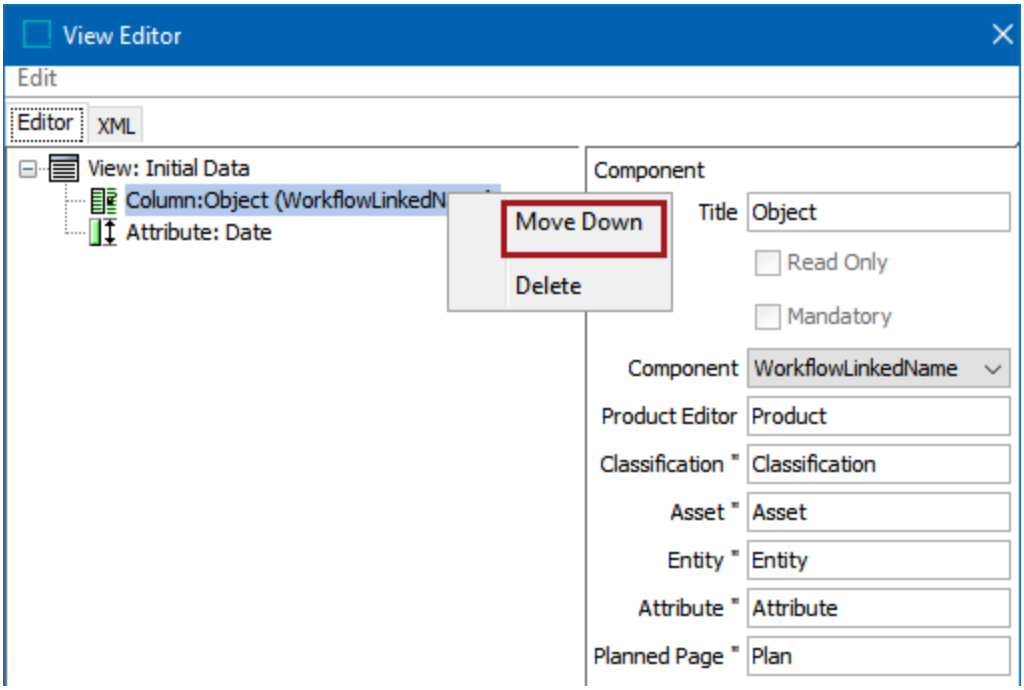
If the option **Sort Columns Automatically** is checked, the Title parameter for each element of the view is used and columns are sorted in alphabetical order based on this value, excluding aspects. To clarify, all columns of the 'Aspect' type (e.g., ID, Name, Revision, etc) appear first in the view, followed by all other elements listed in alphabetical order by Title.





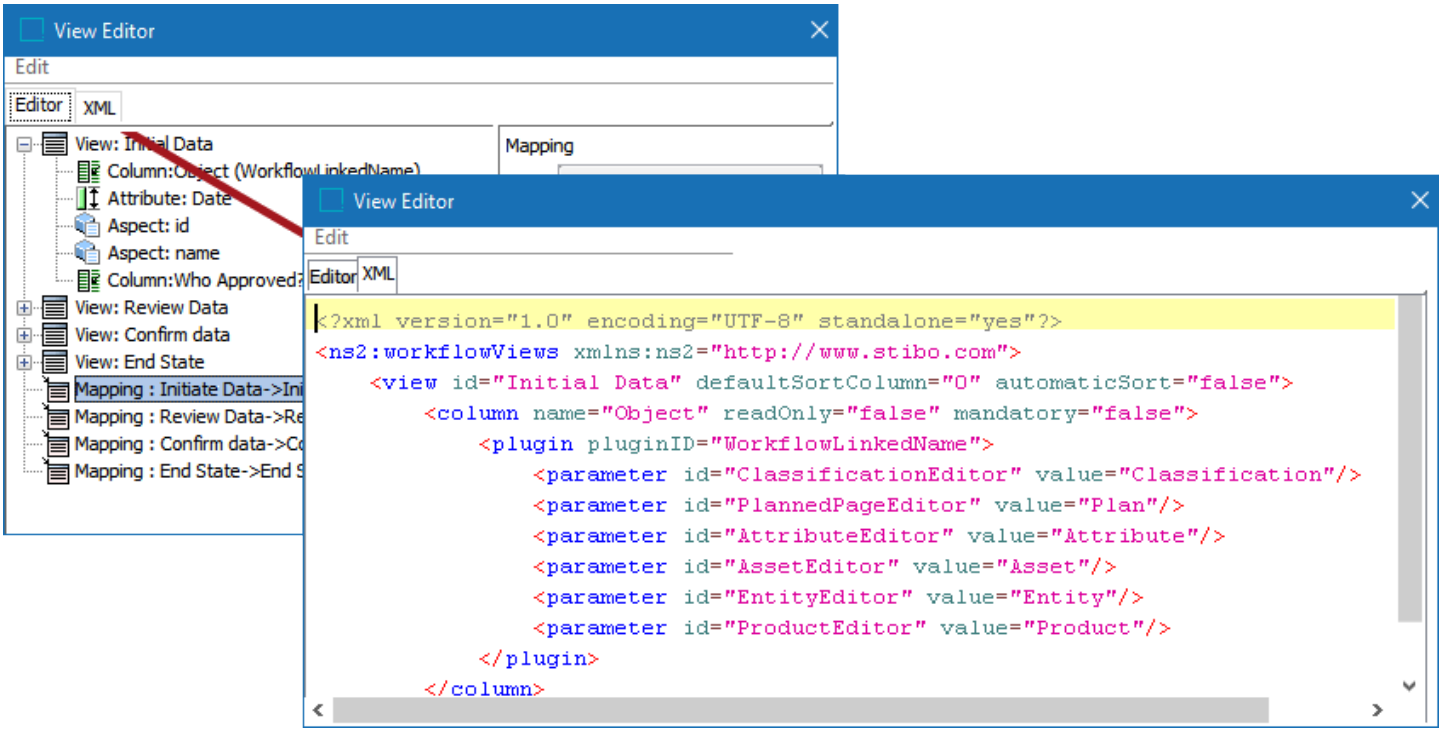
Ordering Columns in Views

It is possible to reorder the data selections in a view by right-clicking on any element of the view and selecting either Move Up or Move Down, depending on what options are given according to the current position. The order of columns in the View Editor determines the order in which they are displayed to the end user so care should be taken in the ordering.



Using the XML View Editor

In addition to the Editor tab on the View Editor dialog, there is also an XML tab. This tab shows an XML representation of the defined views and mappings, allowing users to create and configure views via XML rather than right-clicking. This can be especially useful when a number of similar views need to be configured and workflow designers need to indicate only a few differences between them. In addition, the XML editor can be used to copy views across workflows, which is helpful when multiple workflows use a similar view, but the workflow itself differs and should not be copied (e.g., onboarding and maintenance workflows).



Configuring Mappings for Workflows

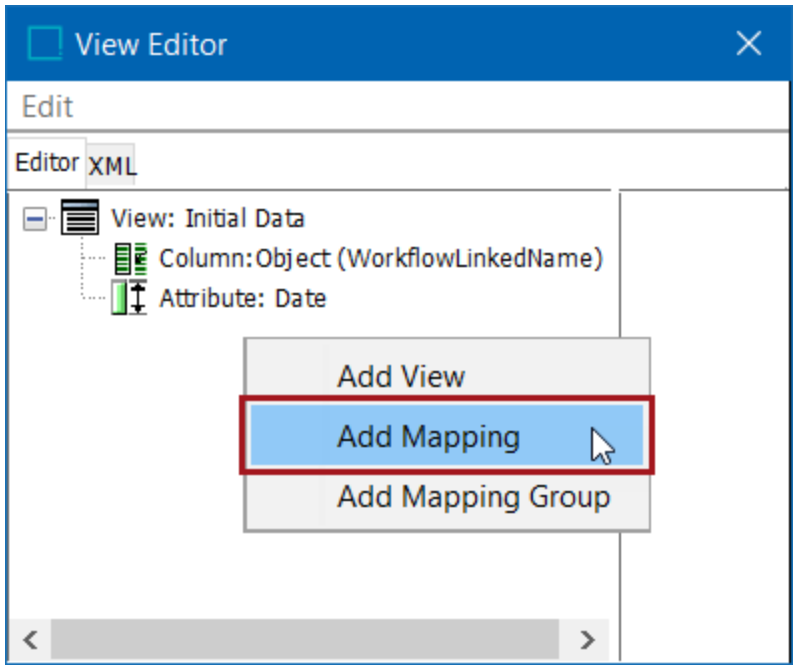
This topic details how to map a view to a workflow state (or states), as well as how to map views to specific users or user groups. Without views and mappings configured, all states in the workflow are displayed on the STEP Workflow navigator tab (including parallels, clusters, and final states). When a workflow includes automated, system, parallel, and/or cluster states, *and* users will be interacting with the workflow in the STEP Workbench, it is recommended to have mappings in place so that only the user states (e.g., states in which a human must interact) are present on the STEP Workflow tab.

Prerequisites

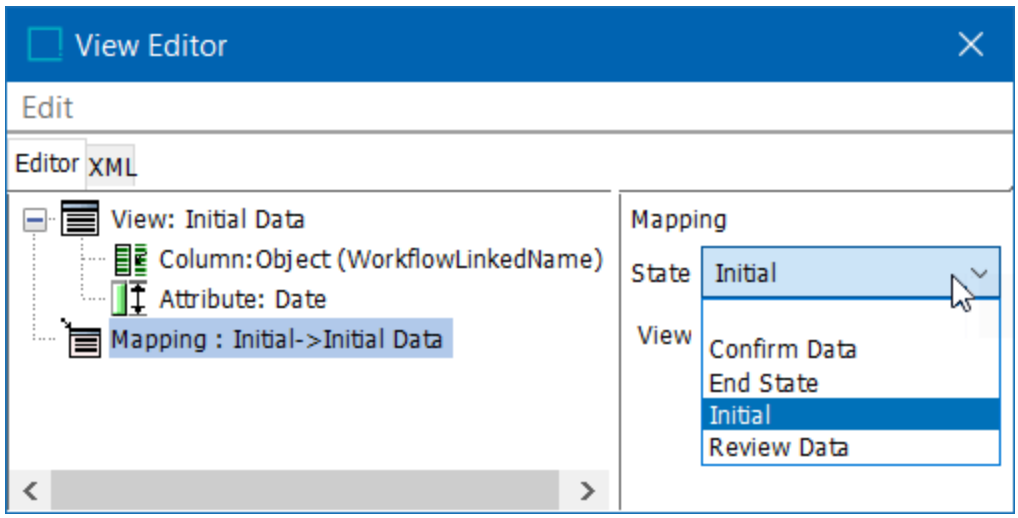
A workflow has been created and one or more views have been created for the workflow. This is required because the mapping task serves to connect any given view to a particular state or set of states. Therefore, for the exercise to be meaningful, both the view(s) and the state(s) must already exist. Also note that the views must have an ID assigned, else they will not be available for selection in the mapping. For more on creating views, see the **Configuring Views for Workflows** topic.

Configuration

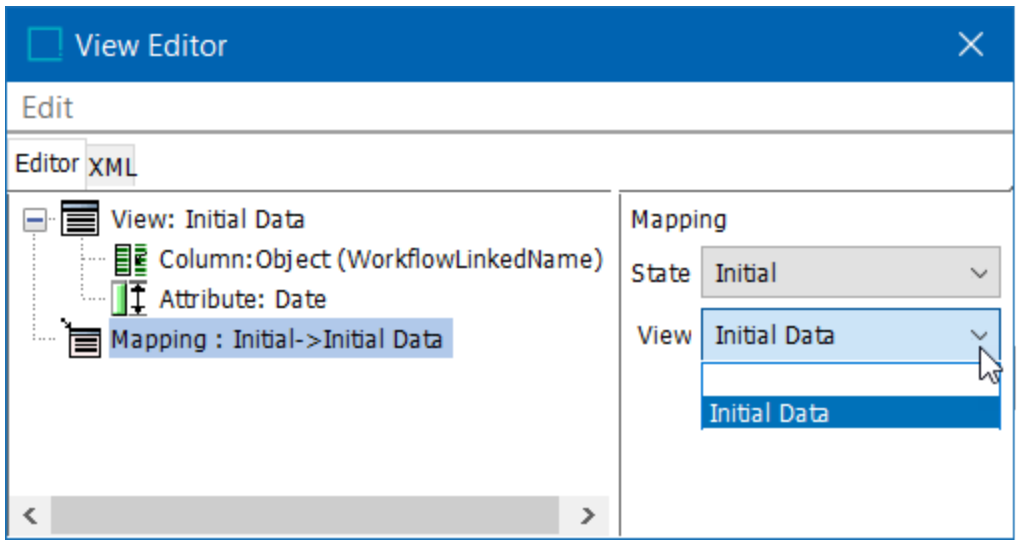
1. To add mapping to a view, right-click the editor panel on the left hand side anywhere, and select **Add Mapping**.



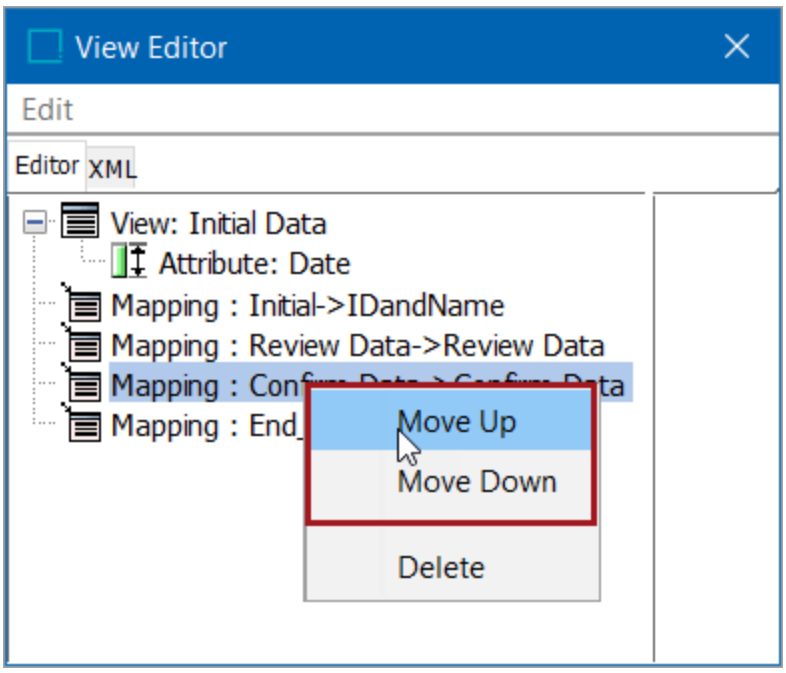
2. In the right-hand panel in the View Editor window that appears, select the state from the dropdown for which you want to apply a view.



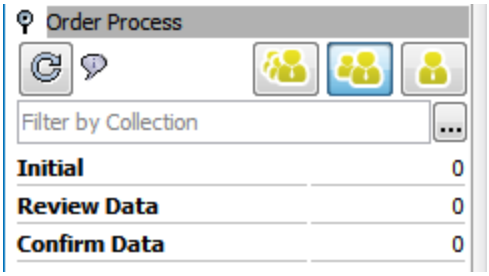
3. From the View dropdown, select the view that you would like to apply to the state.



4. Repeat the above steps as many times as needed so that each state requiring human interaction has a view assigned to it. Note that the same view can be applied to multiple states.
5. Sequence the mappings in the appropriate order by right-clicking on a mapping and selecting 'Move Up' or 'Move down'. Sequencing of views is important as this defines the order in which the views are presented to the user on the STEP Workflow navigator tab. Mapping sequence is typically assigned alphabetically or sequentially (e.g., the first state in the workflow is sequenced first).



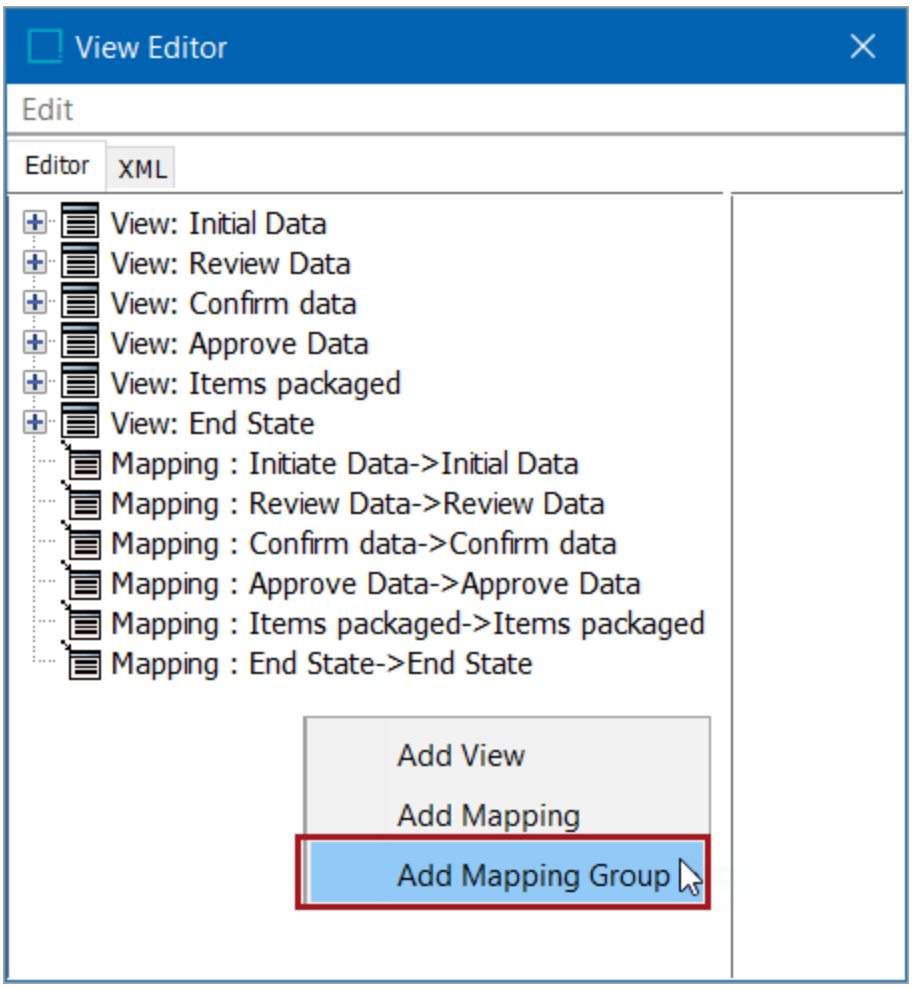
6. Save the workflow, exit the STEP Workflow Designer, and go to the **STEP Workflow** navigator tab.
7. Select the workflow for which view and mapping is being created and expand the flipper. When the flipper is open for the workflow, only the states for which a mapping was created are visible.



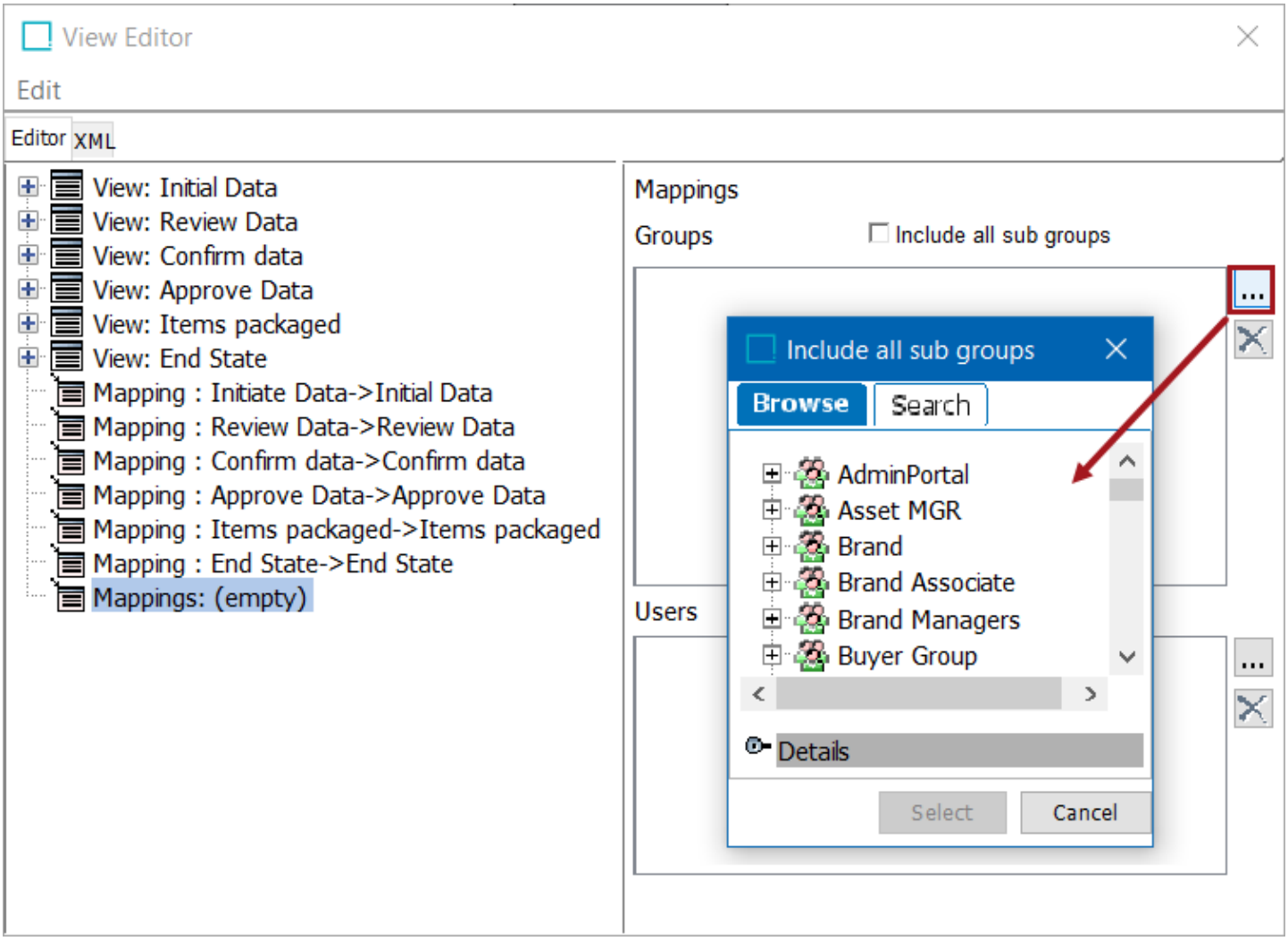
Workflow User / Group Specific Mappings

It is possible to create mappings that only apply to specific users or user groups. The mappings of a particular workflow could be configured so that only users in a specific group can see a given state (e.g., only administrator users have access to automated and system states to intervene in case of errors), or so that users in a specific group could see tasks within the same state with different views (e.g., an end user may need to see different data than an administrator).

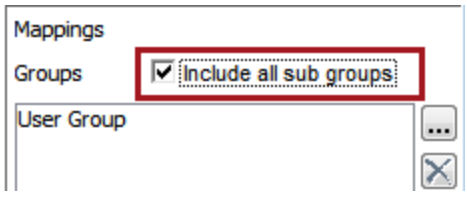
1. To create user / user group specific mappings, right-click the left panel in the View Editor, and select **Add Mapping Group**.



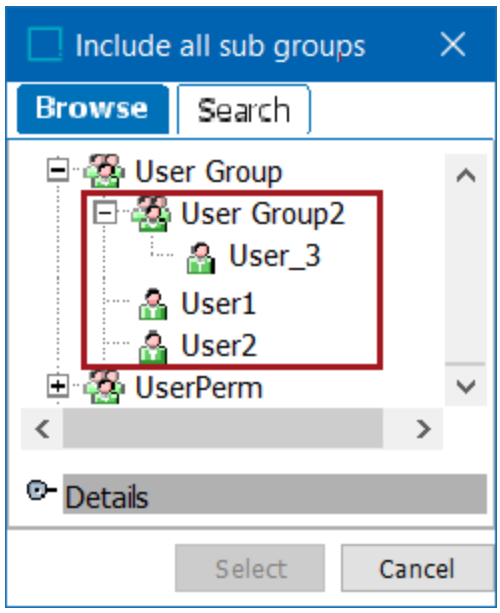
2. Select the user(s) and/or group(s) you want the mapping to apply for. Note that there are separate fields for if a group is added verses a single user.



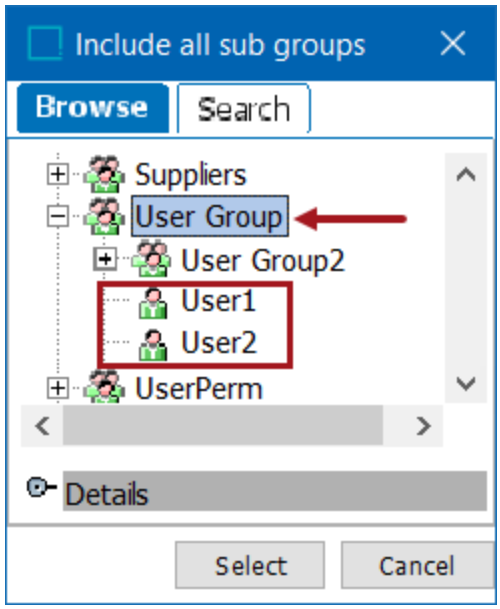
If the **Include all sub groups** is checked, any subgroups within the group, as well as direct users in the group will have access to the view.



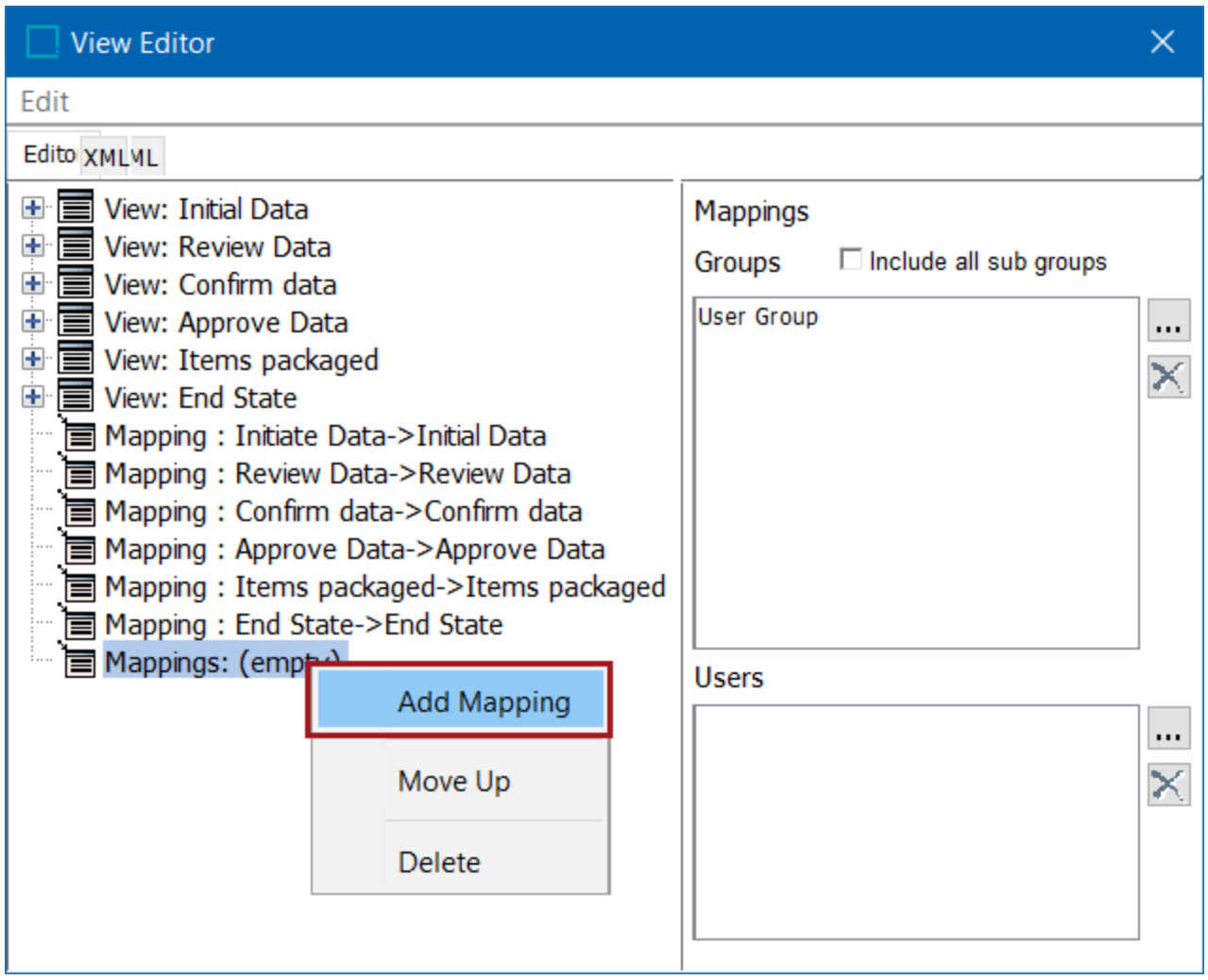
In the example below, if the **Include all sub groups** option was checked and User Group was selected, this would include User Group 2, its child User 3, as well as User 1, and User 2.



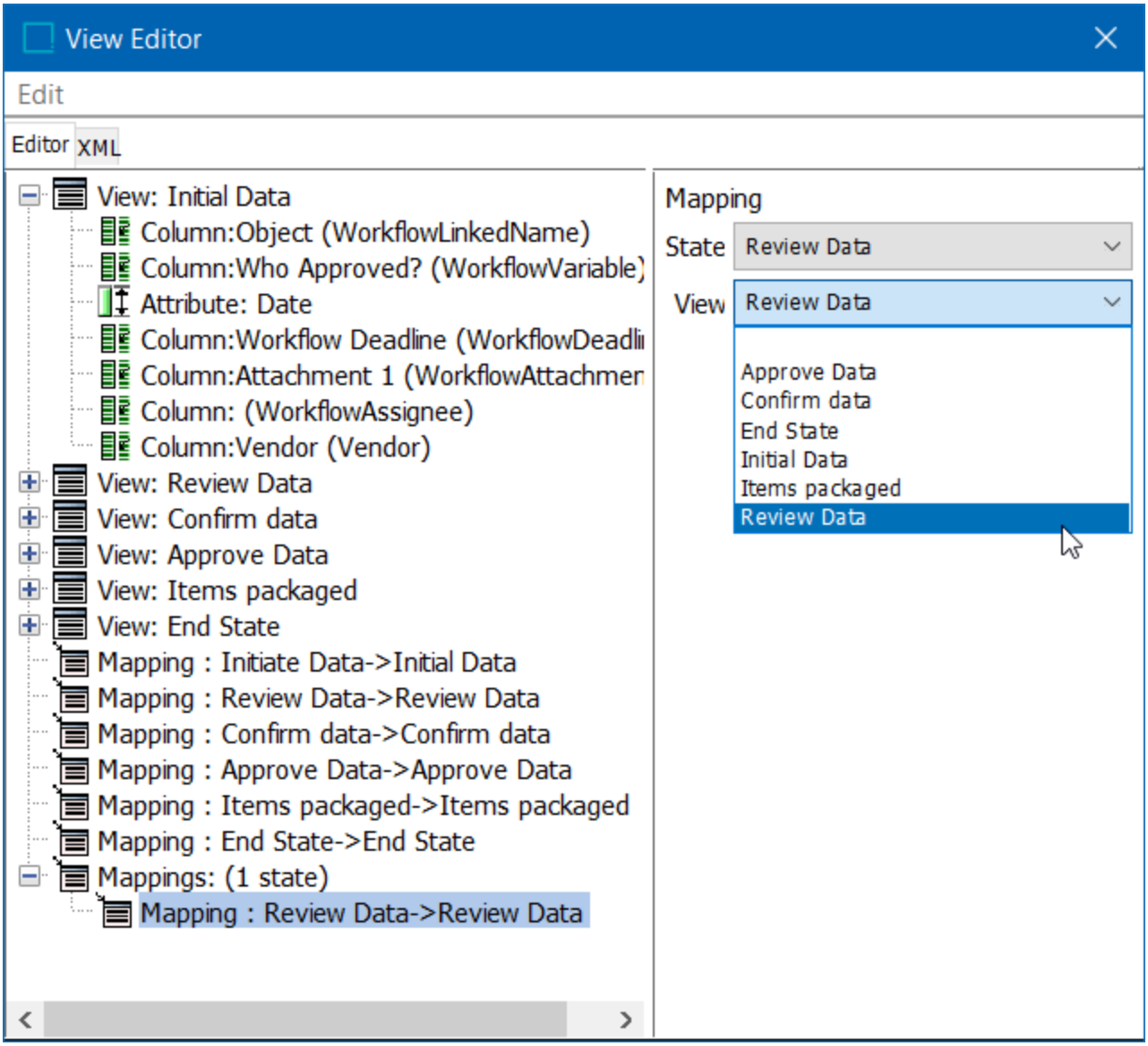
If the **Include all sub groups** box is unchecked, only the top level users directly in the group have access to the view. For example, with User Group selected, only User 1 and User 2 would have access to the view as they are the only direct user children of the selected group.



3. When all users or groups are added, right-click on the **Mappings: empty** node and select **Add Mapping** from the context menu.



4. This will bring up the States and Views dropdown menus to choose from (as with configuring a standard mapping). Select the appropriate state and view for the user or groups to have access to. Close the View Editor and save the workflow.



It is important to note that when a state is mapped to a view both inside and outside a mapping group, and a user logs in who is either directly specified on the mapping group, or member of a group that is specified, the mappings will be evaluated from the top down and the first one that matches will be used. In other words, if the same user is in two different groups with different privileges, but both groups are mapped to the same state, then whatever group comes first in the mapping dictates the allowable privileges for that user. Thus, the order of mappings and mapping groups is significant and mappings must be sequenced appropriately.

Working with Tasks in Workflows

This set of topics focuses on the end user, e.g., the user working with tasks in a workflow in the workbench, rather than the user configuring and/or designing a workflow. After reading these topics, a user should be able to access workflow tasks, edit data on objects in a workflow, and submit tasks through the assigned states in a workflow.

Topics covered in this section are:

- Moving Tasks through a Workflow in Workbench
- The State Log Tab
- Workflow Search Criteria

While concepts remain the same across workbench and Web UI, the interfaces differ. For information specific to Web UI, see the **Workflows in Web UI** section of the **Web User Interfaces** documentation.

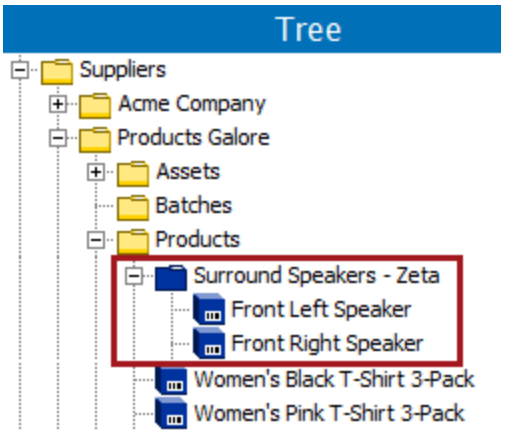
Moving Tasks through a Workflow in Workbench

In order to move an object or task through a workflow in workbench, it first must be initiated into the workflow. This then allows the task to be addressed accordingly through each state in the workflow.

Should a user wish to initiate a task in a workflow in Web UI, see **Moving Tasks through a Workflow in Web UI** in the **Web User Interface** / **Web UI Setup and User Guide** documentation.

Supplier Users

In a workflow, supplier users can only see objects that are **directly** linked to their supplier classification. Objects that are children to blue folders within a classification folder are not available to supplier user workflow tasks. For example, in the following image, the speaker products are not available for workflow tasks because they are child objects to the 'Surround Speakers - Zeta' product folder. The 't-shirt' objects can be worked as tasks because they have a direct link to a classification folder.



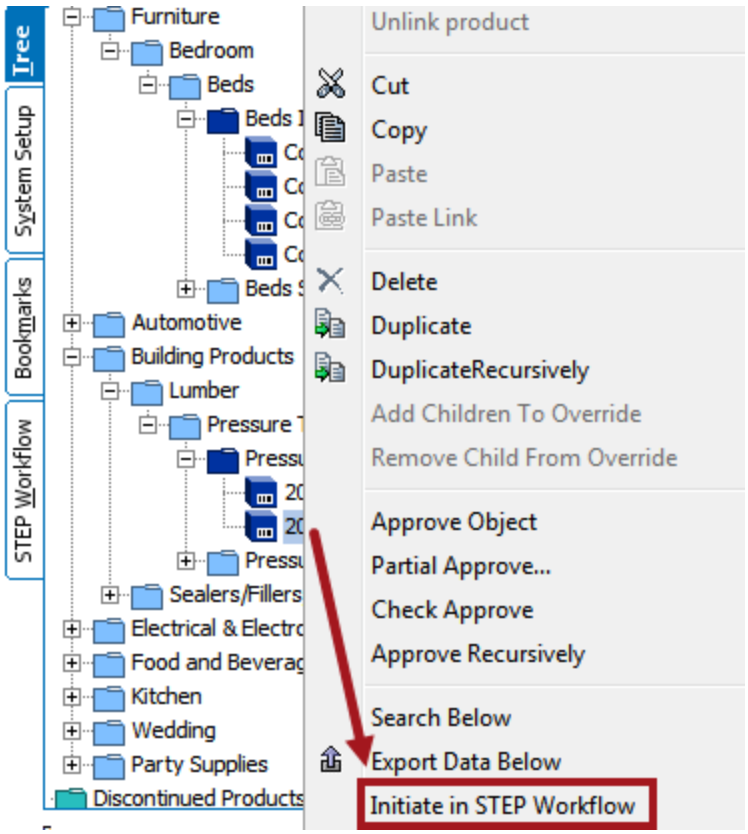
For more information about a supplier item Web UI, see the **Supplier Web UI** topic and the **Supplier Web UI Setup Guide** topic, both in the **Web User Interfaces / Web UI Getting Started** documentation.

Initiating the Object into the Workflow

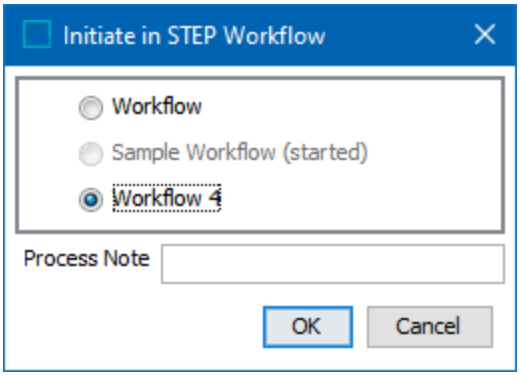
The below process describes how to manually initiate an object into a workflow. However, it is important to note that oftentimes workflows will be configured to automatically initiate objects of a certain type or types into the workflow upon creation of the object. In addition, workflows can be initiated by a variety of other actions or processes (both automated and manually initiated), such as business rules, imports, or bulk updates.

To manually initiate an object into a workflow:

1. Navigate to the **Tree** tab and select the desired object. Right-click and select **Initiate in STEP Workflow**.



2. In the dialog that appears, select which workflow to initiate the object into. Note that only workflows that the object is valid for appear, and any workflows that are disabled indicate that the object has already been initiated into those particular workflows.



If desired, the user can enter a Process Note when starting the workflow. This note is posted to the object's State Log.

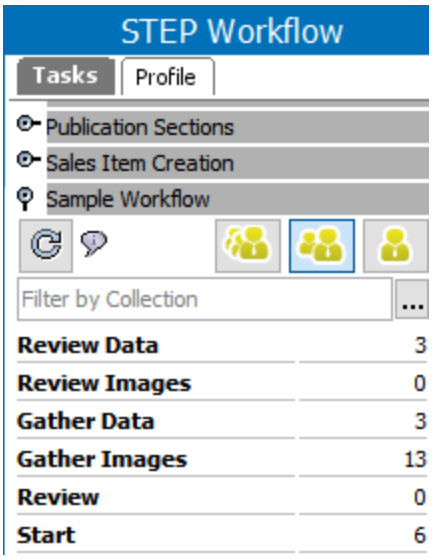
See the **State Log Tab** section of the **Workflows** documentation for more information.

Processing Tasks in a Workflow

Once an object is initiated in a workflow, it is available in one or more states of the workflow. Each instance of an object in a state represents a task to be completed. Each state is assigned to a particular user or group who has the responsibility of completing the tasks in that state. Because of this, tasks are most often accessed via the states themselves, rather than via the objects in the states. The STEP Workflow tab allows users to see tasks in the states to which they have privileges to access. The Tasks tab on an object allows users to see all tasks for a given object, across multiple workflows. Both options are described below.

STEP Workflow Tasks Tab

STEP offers a task list in the workbench that shows all tasks assigned to the current user (either directly to the user or to a group of which the user is a member). This is the primary interface for working with workflows in the workbench and displays all workflows that the user has the 'View and use STEP Workflow' privilege for. When any workflow flipper is expanded, a list of the configured states is displayed, with an object count displaying the number of tasks in the state that the user has access to.



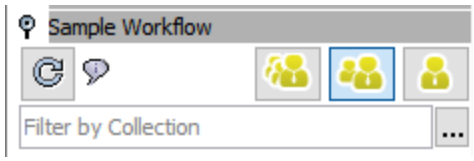
Note that only tasks that are available for the user are displayed. For example, a user who is not part of a group that can process tasks in the Review Data state will see a count of zero (0) rather than three (3).

Note: A legacy 'Advanced' sub tab is available (in addition to the Tasks and Profiles tab shown in the above screenshot). To see the tab, the user must have the 'View STEP Workflow Advanced tab' privilege and the configuration property 'STEPWorkflowNavigationXMLConfigurator.File' must point to a valid configuration file. Such a file is included in the STEP distribution and is unpacked to "STEP_HOME/config-examples/workflow/workbench-navigation.xml" when the application is deployed. If enabled, this tab lets users search for specific tasks based on workflow, state, assignee and a set of configurable search criteria. The tab gives access to see tasks assigned to other users and thus primarily should be used by administrators. Similar functions are available within the standard search functionality for workflow tasks, which honor privileges so it is generally advised to use those instead. For more on using the Search tab see the **Workflow Search**

Criteria topic. In addition, the Profile tab is used for monitoring workflows and is described within the **Monitoring Workflows** section of the **Advanced Workflow Topics** material.

To work with tasks via the STEP Workflow tab:

1. Navigate to the **STEP Workflow** tab, expand the flipper for the desired workflow, and select the appropriate view. There are three options to choose from:



- **Show All (triple user button)**: All tasks are displayed. This button is only available if the user has 'STEP Workflow Administrator' privileges for the workflow.
- **Show Group (double user button)**: Tasks assigned directly to the current user or to a User Group of which he /she is a member are displayed.
- **Show User (single user button)**: Only tasks assigned directly to the current user are displayed.

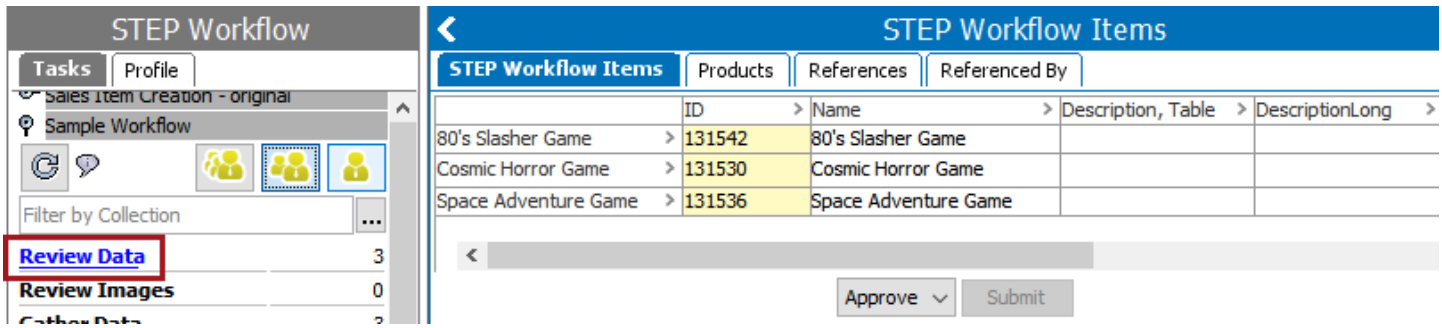
Note: By selecting the Show Group (double user button) icon, users that have the 'View task assigned to other users in my group(s)' privilege are able to view their assigned tasks, as well as any tasks assigned to others and any unclaimed tasks within any group in which they are a member. For more information about workflow privileges, see the Workflows section of the **Setup Actions** topic in the **Action Sets** documentation.

Additionally, a user can further refine the task view by using the 'Filter by Collection' option. If a collection has been previously set up, click the ellipsis button (...) to select that collection will filter the task list to display only tasks for objects contained in the collection. For more on creating collections, see the **Collections** documentation.

2. With the appropriate view selected, click the link for the state in which tasks should be accessed. This will display the tasks in the state according to the view configured by the workflow administrator, allowing users to review and/or edit data as needed. Cells in yellow are read-only, while white cells can be edited. Red cells may be present and are considered mandatory and should be populated prior to submission, though the system may or may not prevent transition of the task if they are not populated (based on how the designer has configured the workflow).

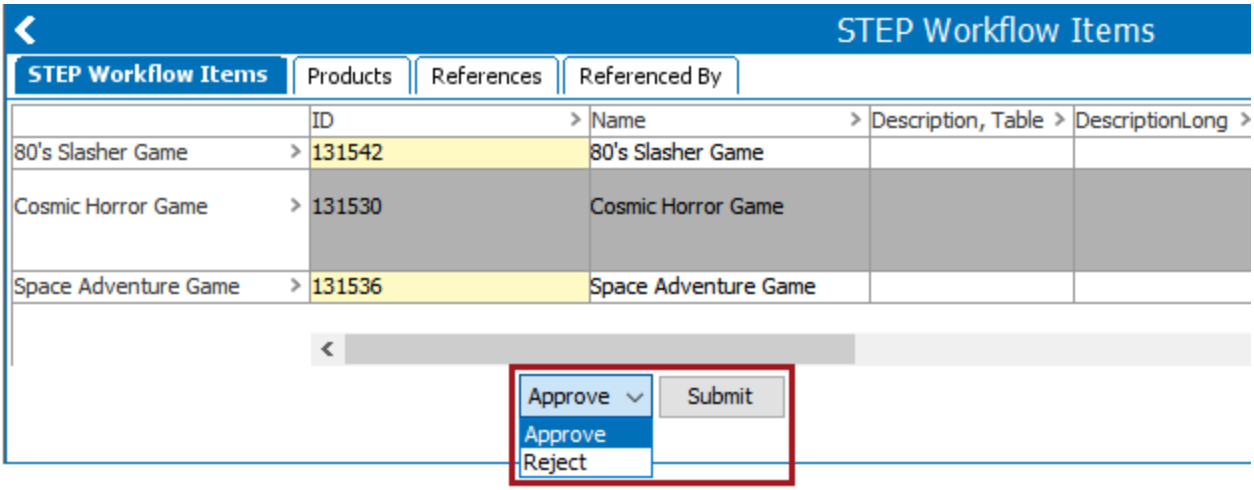
   Some workflows require tasks to be claimed before they can be updated by a user. This can be verified by the presence / absence of a task claiming icon ( ). If this is present in one of the columns on the STEP Workflow Items tab, see the **Claiming and Releasing Tasks in Workflows** section of the **Advanced Workflow Topics** material for additional information.

   Note: If it is preferred to see in the objects listed in columns instead of rows in the STEP Workflow Items editor, right-click the table header and select 'Rotate Table'.

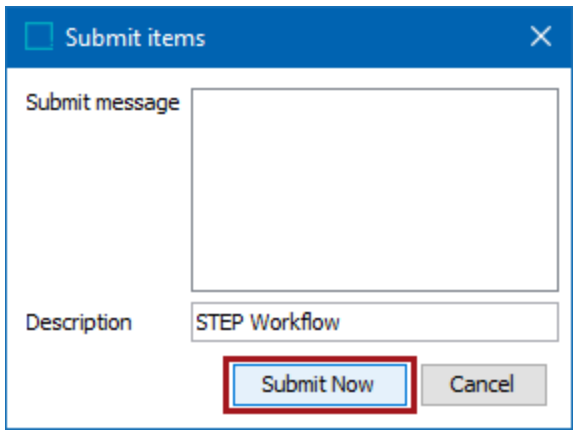


Note: The STEP Workflow Items editor will as a default show a maximum of 100 tasks / objects when you select a state on the Tasks sub tab. This limit can however be changed by a system administrator via the config property "STEPWorkflow.Assignee.TaskList.Max".

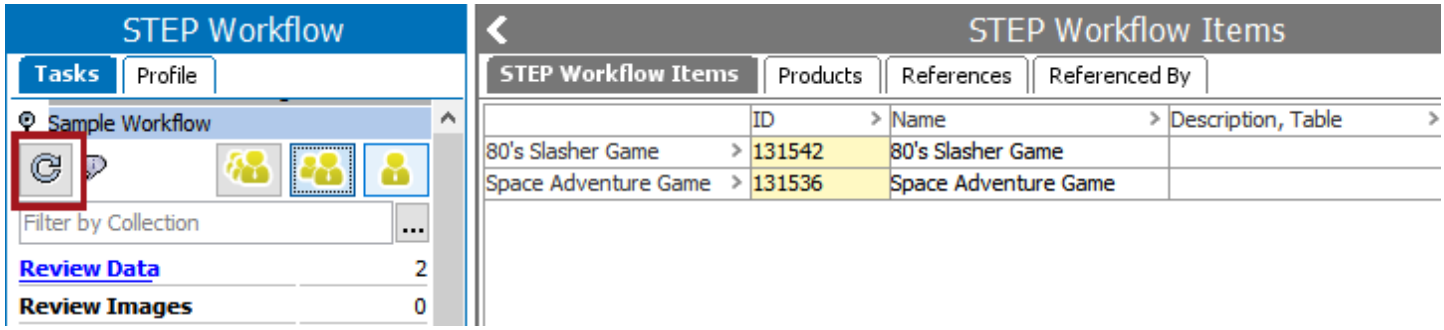
3. Submitting a task to a subsequent state indicates that work on the task is complete. Unless the user has access to a subsequent task for that object, it will no longer be accessible for them in the workflow. To submit a task, select the task (multi-select is enabled using Ctrl + click), choose the appropriate selection from the dropdown (if applicable) and click Submit. The available options for selection are configurable based on the design of the workflow. In some cases only a single selection will be present and therefore it is not required for the user to select a specific transition.



4. A dialog will appear, allowing the user to input a comment, which will be recorded in the Notes field on the State Log tab of the object. Note that if multiple objects were selected, the same comment will be applied to all. From the dialog, click the Submit Now button to complete the task, or click Cancel to return to the STEP Workflow Items editor.

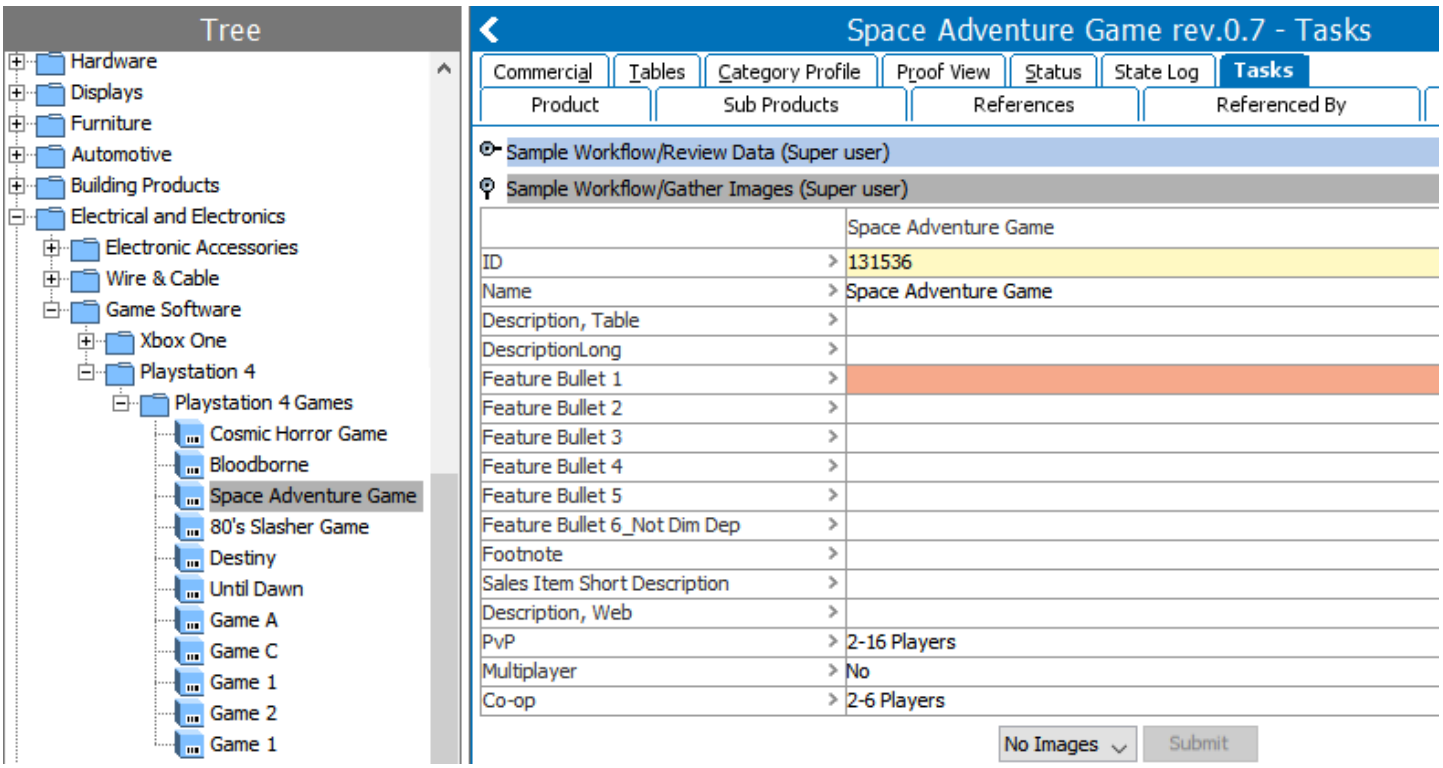


5. Clicking the Refresh button in the Task pane will update the task list.



Tasks Tab on an Object

From an object in Tree, the Tasks tab displays all tasks across all workflows for that object, subject to the user's privileges. Only tasks that the user has rights to address are visible.



The interface and interactions are comparable to that of the task list, with the only difference being that users can expand or collapse flippers to access tasks in various states and workflows.

State Log Tab

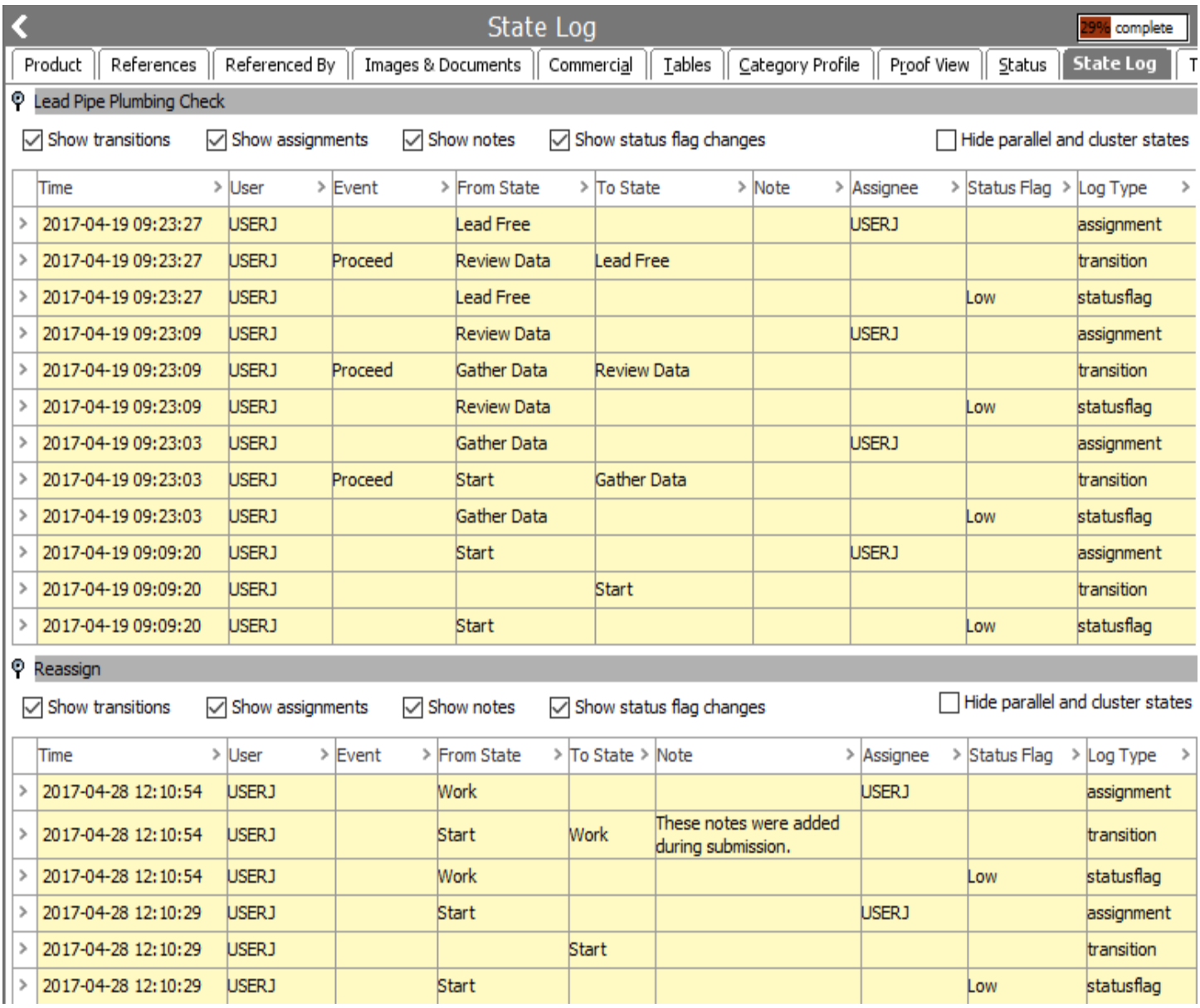
Any object type for which a workflow can be configured has an editor tab named State Log. This is useful for tracking the recent history of an object across all workflows that include the object. Initiating an object into a new workflow creates a flipper under which recent history is housed for that particular workflow.

All transitions, assignments, notes, and status flags are displayed with a row in the table. For each entry there is a timestamp, a log type, and the name of the user causing the entry. Note that only the latest 50 entries per workflow are displayed.

The following checkboxes filter the state log, allowing a user to view only the necessary information for the object in the workflow indicated by the flipper. At least one checkbox must be checked for the workflow information to be displayed.

- **Show transitions**: when checked, the 'transition' log type entry is displayed, adding data for the 'To State' column, in addition to the 'From State' column.
- **Show assignments**: when checked, the 'assignment' log type entry is displayed in the 'Assignee' column, showing the user to whom the task has been assigned.
- **Show notes**: when checked, any notes entered while transitioning from one state to another are displayed in the 'Note' column.
- **Show status flag changes**: when checked, the 'statusflag' log type entry is displayed, adding data for the 'Status Flag' column.
- **Hide Parallel and Cluster States**: when checked, any states in a workflow which has Parallel or Cluster states are displayed. Since entering and leaving parallels and clusters result in a lot of entries, this option allows you to reduce the number of rows displayed for those types of workflows.

The log does not reset with every new workflow instance. Thus, if an object has been removed from a workflow and then later initiated into the same workflow again, entries from the first instance continue to display.



State Log

29% complete

Product | References | Referenced By | Images & Documents | Commercial | Tables | Category Profile | Proof View | Status | State Log | T

Lead Pipe Plumbing Check

☑ Show transitions ☑ Show assignments ☑ Show notes ☑ Show status flag changes ☐ Hide parallel and cluster states

Time	User	Event	From State	To State	Note	Assignee	Status Flag	Log Type
2017-04-19 09:23:27	USERJ		Lead Free			USERJ		assignment
2017-04-19 09:23:27	USERJ	Proceed	Review Data	Lead Free				transition
2017-04-19 09:23:27	USERJ		Lead Free				Low	statusflag
2017-04-19 09:23:09	USERJ		Review Data			USERJ		assignment
2017-04-19 09:23:09	USERJ	Proceed	Gather Data	Review Data				transition
2017-04-19 09:23:09	USERJ		Review Data				Low	statusflag
2017-04-19 09:23:03	USERJ		Gather Data			USERJ		assignment
2017-04-19 09:23:03	USERJ	Proceed	Start	Gather Data				transition
2017-04-19 09:23:03	USERJ		Gather Data				Low	statusflag
2017-04-19 09:09:20	USERJ		Start			USERJ		assignment
2017-04-19 09:09:20	USERJ			Start				transition
2017-04-19 09:09:20	USERJ		Start				Low	statusflag

Reassign

☑ Show transitions ☑ Show assignments ☑ Show notes ☑ Show status flag changes ☐ Hide parallel and cluster states

Time	User	Event	From State	To State	Note	Assignee	Status Flag	Log Type
2017-04-28 12:10:54	USERJ		Work			USERJ		assignment
2017-04-28 12:10:54	USERJ		Start	Work	These notes were added during submission.			transition
2017-04-28 12:10:54	USERJ		Work				Low	statusflag
2017-04-28 12:10:29	USERJ		Start			USERJ		assignment
2017-04-28 12:10:29	USERJ			Start				transition
2017-04-28 12:10:29	USERJ		Start				Low	statusflag

Workflow Search Criteria

The STEP Workflow Search criteria lets a user search for:

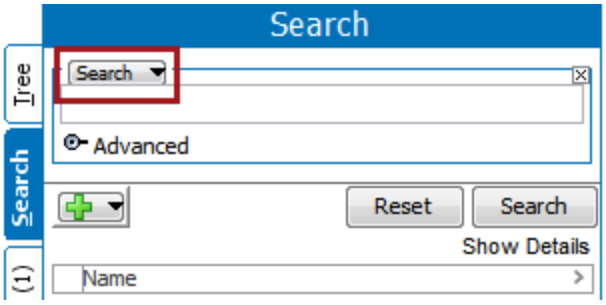
1. Objects that are in a specific state in a specific workflow
2. Objects that are in a specific workflow but could be in any state
3. Objects that are valid for a given workflow, but are not in it

These options are helpful because they allow users to interact with workflows more efficiently. For example, if a user wants to update a group of items that are already in a specific workflow, they can use search to identify the items in a particular state and perform a bulk update on all objects. Additionally, if a user wants to see which items are valid for a particular workflow and which have not yet been initiated they can do so. All of these options assist users when working with workflows.

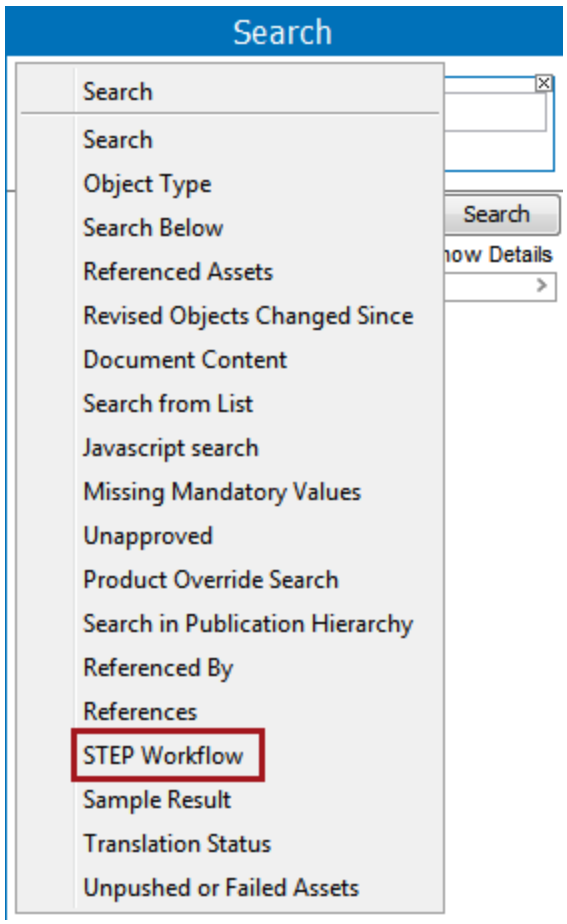
Performing a Search with Workflows

Below is an example of how the Search tab can be used for workflows.

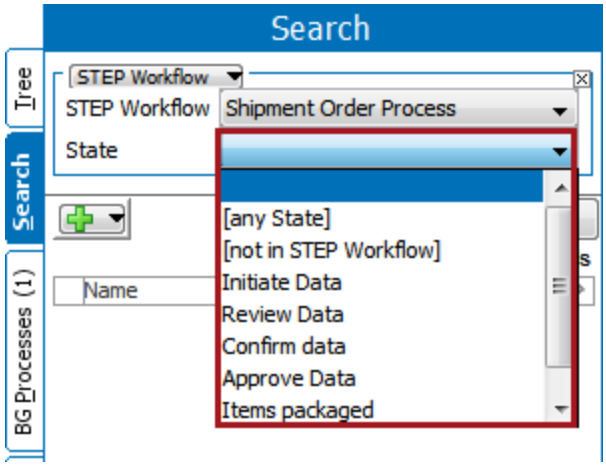
1. Navigate to the side Search tab and click on it. A dropdown is present to specify search criteria.



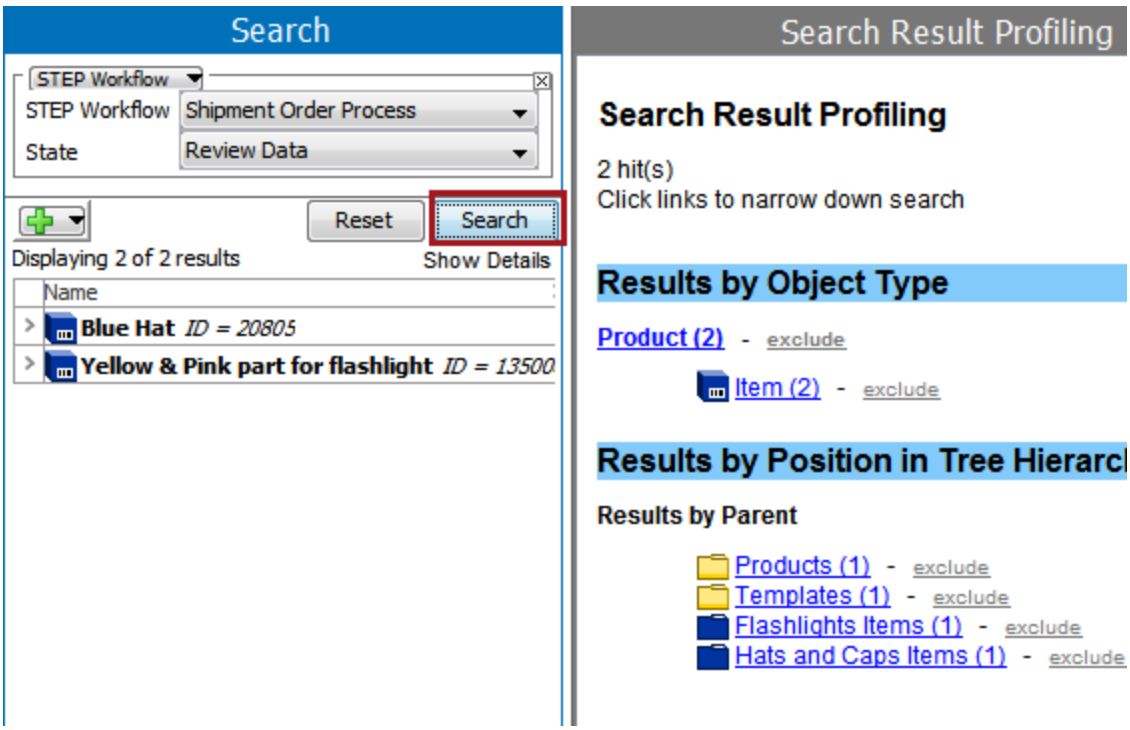
2. Click on the dropdown and select **STEP Workflow**.



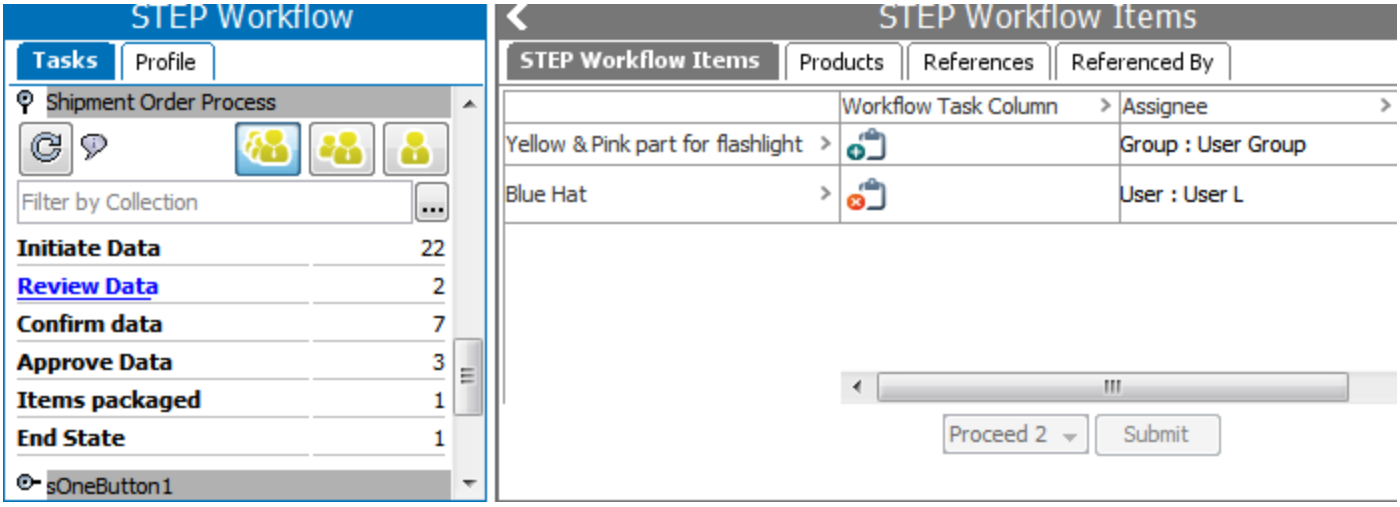
3. When the STEP Workflow criteria is selected, two parameters are available for a user to specify workflow and state. A workflow must be selected, but selection of a state is optional. Also note the bracketed options in the State parameter - allowing for users to identify all objects currently in the workflow, or all objects valid for the workflow but not currently active in it.



4. When the options have been selected, click on Search to run the search and view any results.



If a user is checking to see what tasks are in a state, they can also verify the results by going to the Tasks sub tab in the STEP Workflow tab. See the **Moving Tasks through a Workflow in Workbench** section of the **Workflows** documentation for more information.



5. If the search is not as refined as needed, a user can add additional search criteria by pressing the green + button, , and further filtering the results. For example, the workflow criteria could be combined with the Search Below criteria to isolate objects in a particular category.

For more information on the various search criteria and/or the actions that can be taken with a set of search results, see the **Search** and child topics in the **Getting Started / User Guide** documentation.

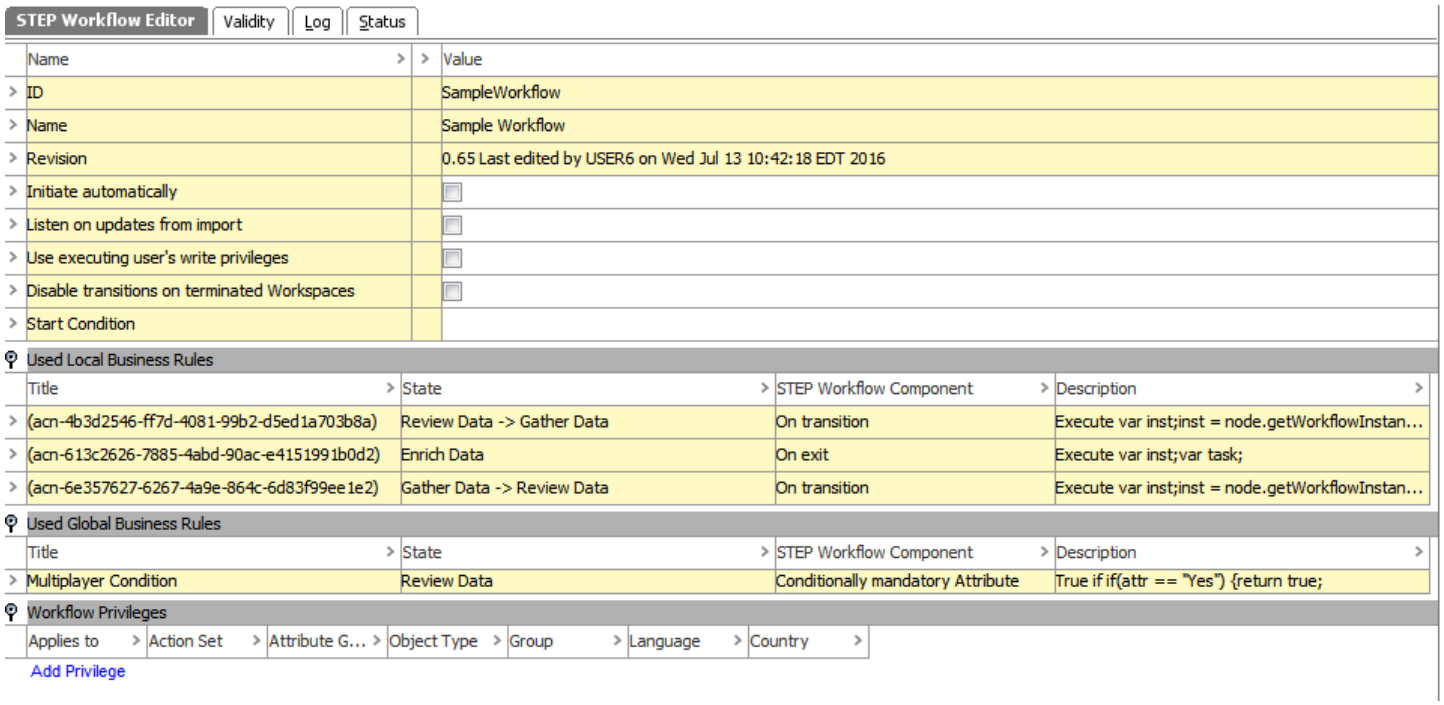
Managing Workflows

When a workflow is selected from the System Setup tab, various settings for the workflow can be viewed and/or edited (if applicable) on the different tabs. These settings help in giving valuable information in regards to the particular workflow such as an overview of the workflow, what the workflow is valid for, a list of users who have interacted with the workflow or made changes, and any revision history.

To view the workflow tabs, go to System Setup, expand the workflows node, and select the relevant workflow. This opens the workflow in an editor. Note that workflows viewed in the Approved Workspaces are read-only.

STEP Workflow Editor Tab

The STEP Workflow Editor tab displays basic information about the workflow, as well as providing some global configuration options for the workflow.

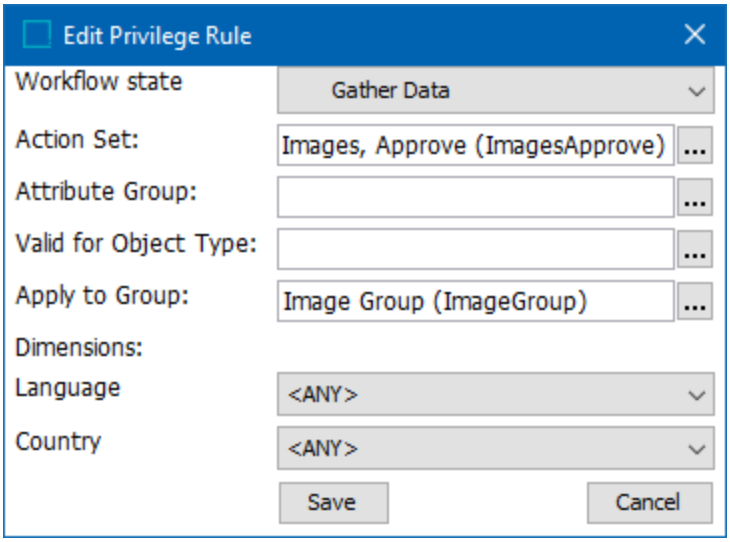


For more information on the basic settings, see **Creating a Workflow** in the **Workflows** documentation.

The STEP Workflow Editor tab also has two flipper areas that show an overview of the business rules used in a workflow: 'Used Local Business Rules' and 'Used Global Business rules'.

The Workflow Privileges flipper displays state-specific privileges for the workflow, which can be edited from the Privilege Rules tab on a User Group, or from this interface. This functionality is useful when a particular user group should have access to data *only when an object is in the indicated state.* For example, a user group may have rights to *edit* certain data only when accessing the data from a particular state or an onboarding / creation workflow or a maintenance workflow, but can only *view* the data for objects not in the editable workflow states.

To edit the workflow privileges, click on the Add Privilege link, and select the state that the particular privilege is to apply to from the drop-down menu in the pop-up window.

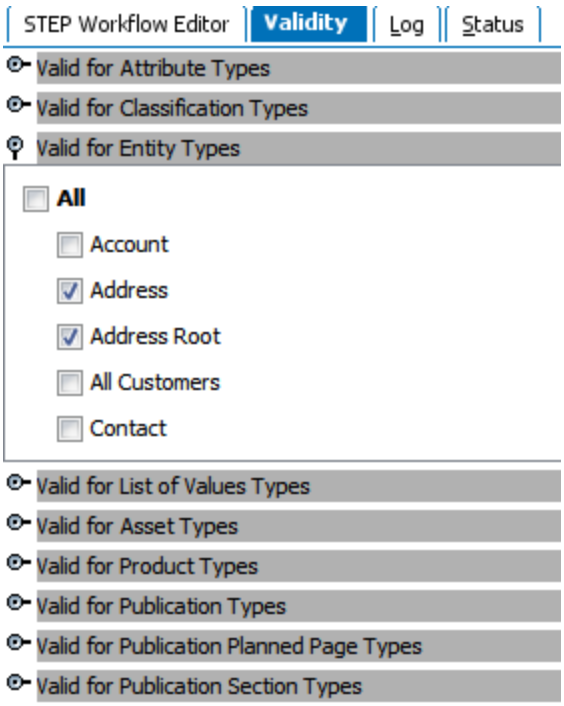


Once the state has been chosen for which the privileges should be applied, it is then possible to set up the various options to set the privileges for that particular state. Complete configuration information for workflow privilege rules is provided in the **Editing Privilege Rules** section of the **System Setup / Super User Guide** documentation.

Note that if workflow privileges are applied to a state, a 'P' indicator will appear on the state within the workflow editor: 

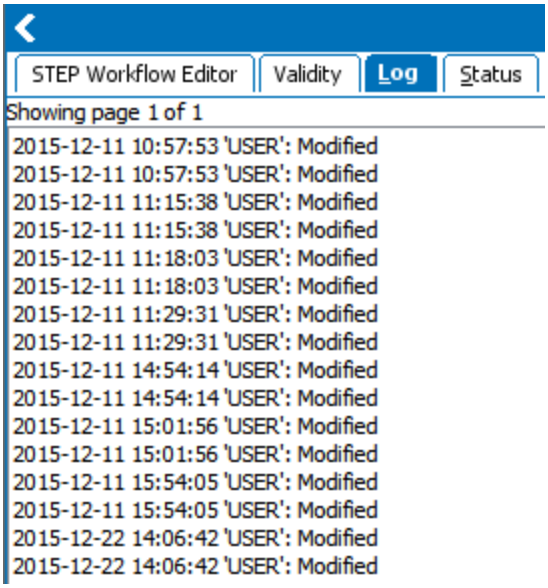
Validity Tab

The Validity tab allows for specification of which object types the workflow is valid for. Only objects of the selected types can be initiated into the workflow. Notice that while a workflow can be made valid for multiple object types of a specific node type, it cannot be valid for object types of different node types. For example, a workflow can be valid for multiple types of entities, but not for both entities and classifications.



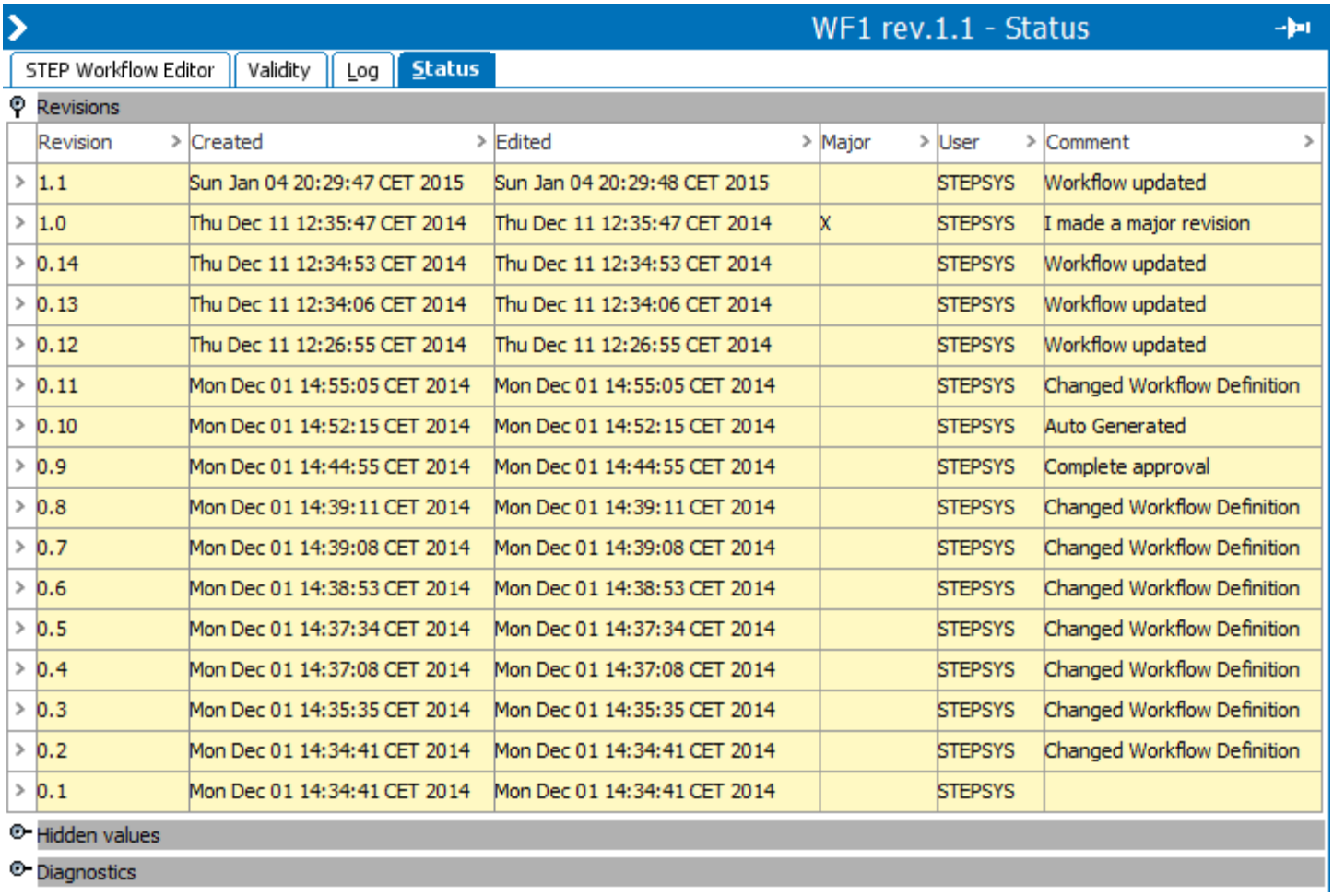
Log Tab

The Log tab is a list of dates and times when changes were made to the workflow. Listed with each time stamp is the user who made each change.



Status Tab

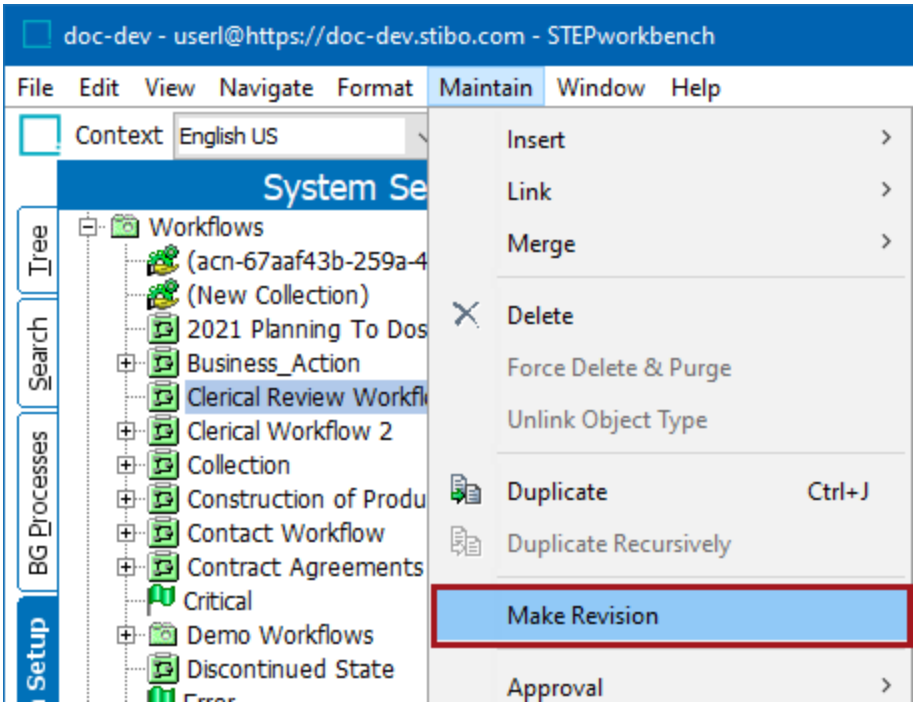
The Status tab provides an overview of the revisions of the workflow, including the user who changed the object and when the change occurred.



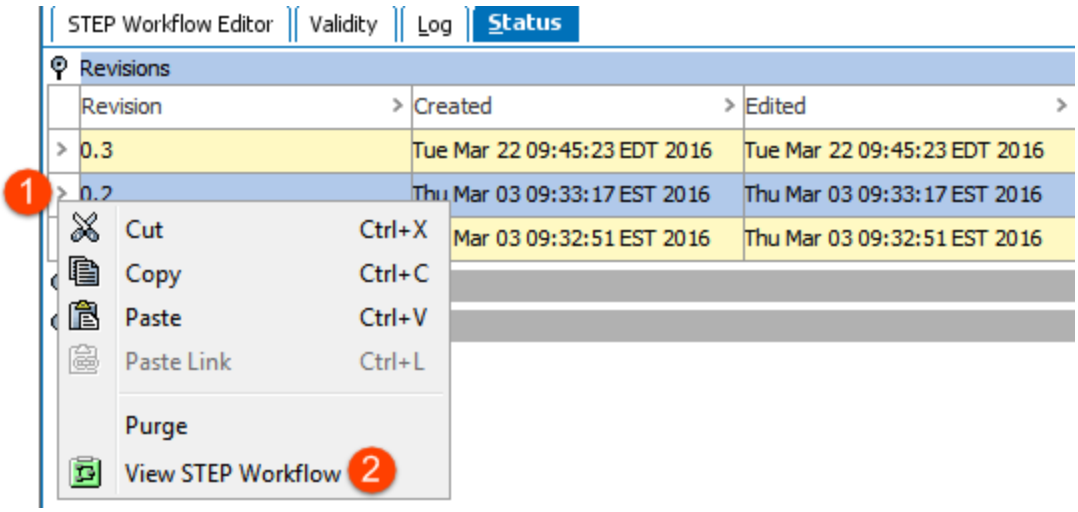
Revision	Created	Edited	Major	User	Comment
1.1	Sun Jan 04 20:29:47 CET 2015	Sun Jan 04 20:29:48 CET 2015		STEPSYS	Workflow updated
1.0	Thu Dec 11 12:35:47 CET 2014	Thu Dec 11 12:35:47 CET 2014	X	STEPSYS	I made a major revision
0.14	Thu Dec 11 12:34:53 CET 2014	Thu Dec 11 12:34:53 CET 2014		STEPSYS	Workflow updated
0.13	Thu Dec 11 12:34:06 CET 2014	Thu Dec 11 12:34:06 CET 2014		STEPSYS	Workflow updated
0.12	Thu Dec 11 12:26:55 CET 2014	Thu Dec 11 12:26:55 CET 2014		STEPSYS	Workflow updated
0.11	Mon Dec 01 14:55:05 CET 2014	Mon Dec 01 14:55:05 CET 2014		STEPSYS	Changed Workflow Definition
0.10	Mon Dec 01 14:52:15 CET 2014	Mon Dec 01 14:52:15 CET 2014		STEPSYS	Auto Generated
0.9	Mon Dec 01 14:44:55 CET 2014	Mon Dec 01 14:44:55 CET 2014		STEPSYS	Complete approval
0.8	Mon Dec 01 14:39:11 CET 2014	Mon Dec 01 14:39:11 CET 2014		STEPSYS	Changed Workflow Definition
0.7	Mon Dec 01 14:39:08 CET 2014	Mon Dec 01 14:39:08 CET 2014		STEPSYS	Changed Workflow Definition
0.6	Mon Dec 01 14:38:53 CET 2014	Mon Dec 01 14:38:53 CET 2014		STEPSYS	Changed Workflow Definition
0.5	Mon Dec 01 14:37:34 CET 2014	Mon Dec 01 14:37:34 CET 2014		STEPSYS	Changed Workflow Definition
0.4	Mon Dec 01 14:37:08 CET 2014	Mon Dec 01 14:37:08 CET 2014		STEPSYS	Changed Workflow Definition
0.3	Mon Dec 01 14:35:35 CET 2014	Mon Dec 01 14:35:35 CET 2014		STEPSYS	Changed Workflow Definition
0.2	Mon Dec 01 14:34:41 CET 2014	Mon Dec 01 14:34:41 CET 2014		STEPSYS	Changed Workflow Definition
0.1	Mon Dec 01 14:34:41 CET 2014	Mon Dec 01 14:34:41 CET 2014		STEPSYS	

Revisions are created when the workflow has had edits made to it in the STEP Workflow Designer, and the designer is saved and closed. Minor revisions are made automatically when various users interact with the same object.

Major revisions can be made on both the object initiated in the workflow or the workflow itself. It is recommended to make a major revision for a workflow each time a set of changes has completed. For example, it may be useful to make a major revision when UAT begins for a workflow so it is clear what is being tested, and again upon completion of UAT so that it is clear what has been released for production. To do so, click on the workflow in System Setup, and select Make Revision from the Maintain menu.



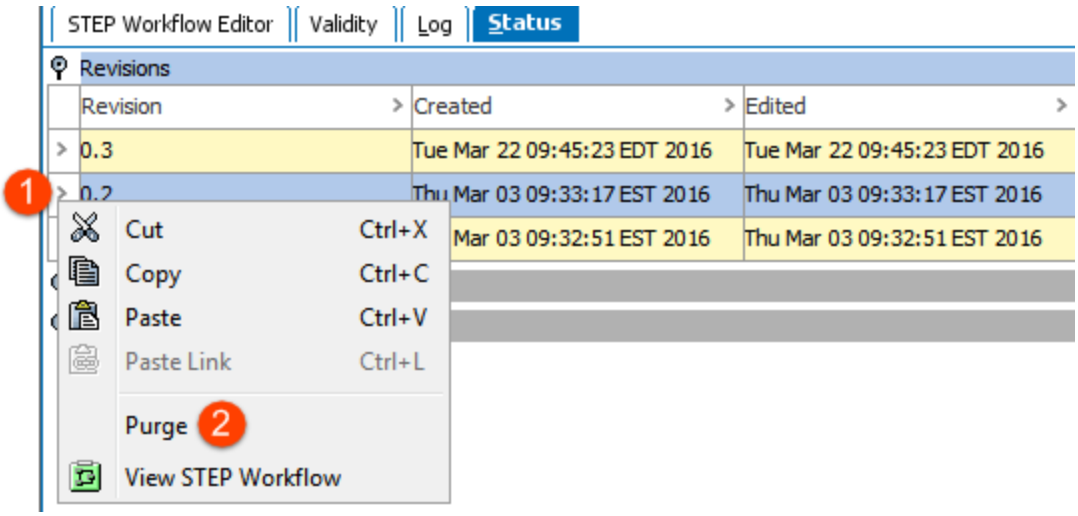
Previous versions of STEP Workflows can be viewed by selecting the revision on the Status tab, right-clicking on the row indicator (>), and selecting View STEP Workflow.



Purging Workflow Revisions

Any revision can be purged, but once a revision is purged it cannot be recovered.

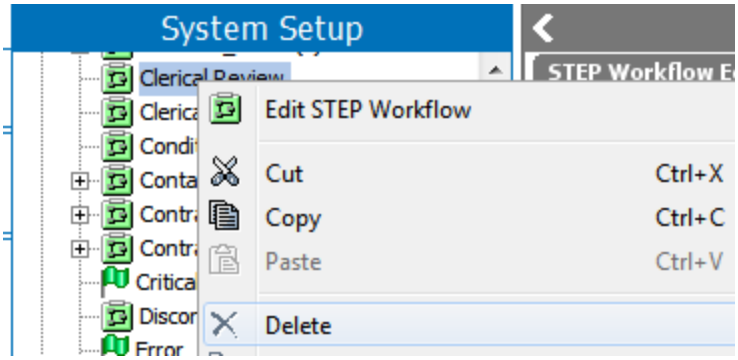
To purge a revision, right-click on the row indicator (>) next to the desired revision and select Purge.



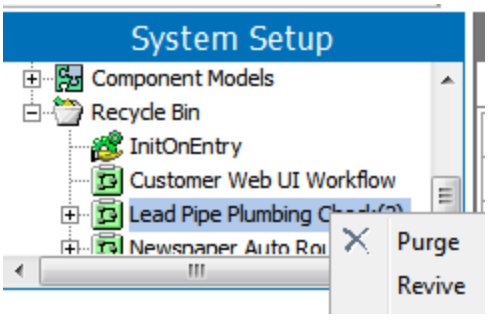
For more information on the Status Tab and revisions, see **Revisions** in the **System Setup / Super User Guide** documentation.

Deleting Workflows

To delete an entire workflow, right-click on it from the System Setup tab and select Delete.



Deleted workflows will be available in the System Setup Recycle Bin until they are purged (permanently deleted) or revived (restored to their previous location).



Advanced Workflow Topics

This portion of the material covers more advanced workflow configurations, going beyond the basic setups described in the introductory material. Users should be familiar with all material in the **Getting Started with STEP Workflows** documentation before attempting to configure and/or utilize more advanced workflow functionality.

Most real-world workflows will utilize many of the functions described in the advanced topics, but none are required in any particular workflow. The topics generally do not build on one another and completing one configuration typically does not require knowledge of any other, outside of the basic workflow setup.

In other words, the advanced workflow documentation is function and topic based, meaning that each topic describes a particular function a user may wish to carry out (e.g., send email from a workflow or remove objects from a workflow) or a particular topic a user may wish to read about (e.g., workflow attachments or workflow deadlines).

Therefore, there is no suggested reading order and users should approach the material in the order that seems most relevant for their workflow.

Assignees in Workflows

In workflows, an object cannot be in a state without an assignee being set. This regardless of whether the state represents a human task or not. The assignee must be an existing STEP User or user group.

An important thing to notice when working with assignees is that users who do not have the 'STEP Workflow Administrator' privilege for a given workflow only will be able to see tasks assigned to themselves or a group they are a member of. Furthermore, they will only be able to work with tasks assigned directly to themselves and have to claim a task (i.e., assign it to themselves) to be able to edit workflow related data and perform transitions. Users with the 'STEP Workflow Administrator' privilege can view and work with all tasks regardless of who they are assigned to.

Note: For states that do not represent a human task, you will typically want the assignee to be either an administrator with in-depth knowledge about the workflow or a dedicated 'automation' dummy user.

There are multiple ways of setting the assignee for a state. In the sections below, the options in the Workflow Designer State Editor are described.

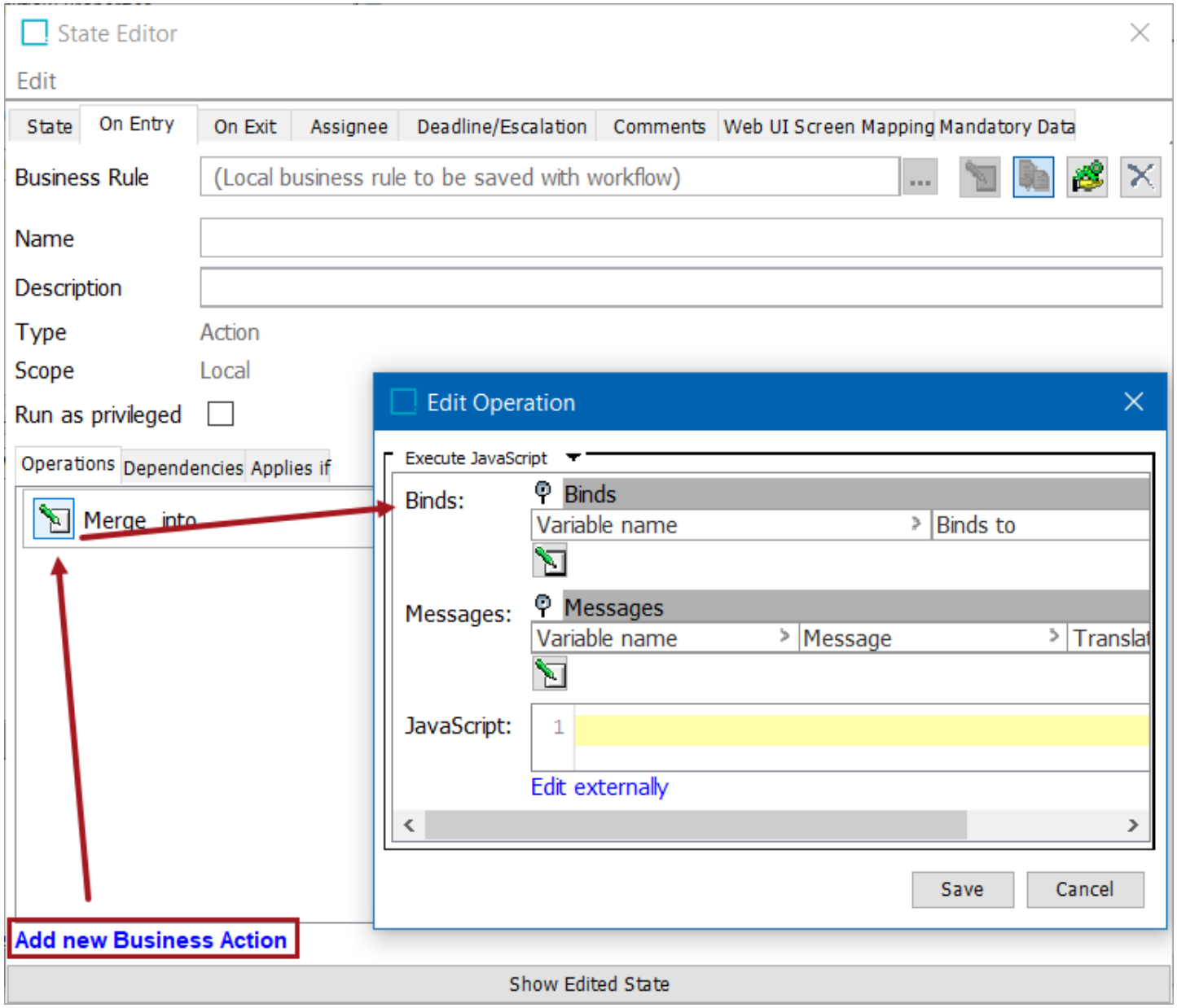
For setting the assignee using the On Entry tab in the State Editor, see the **Assigning Tasks Using Business Rules** topic in the Advanced Workflow Topics section of the **Workflows** documentation.

For more on retrieving user IDs who work with tasks in workflows, see the **Retrieving the ID of an Assignee Completing a Task in a Workflow** in the Advanced Workflow Topics section of the **Workflows** documentation.

Note: Attempting to delete a user who an assignee for a workflow state will result in errors. For more information, see the **Working with Users** topic in the **System Setup / Super User Guide** documentation.

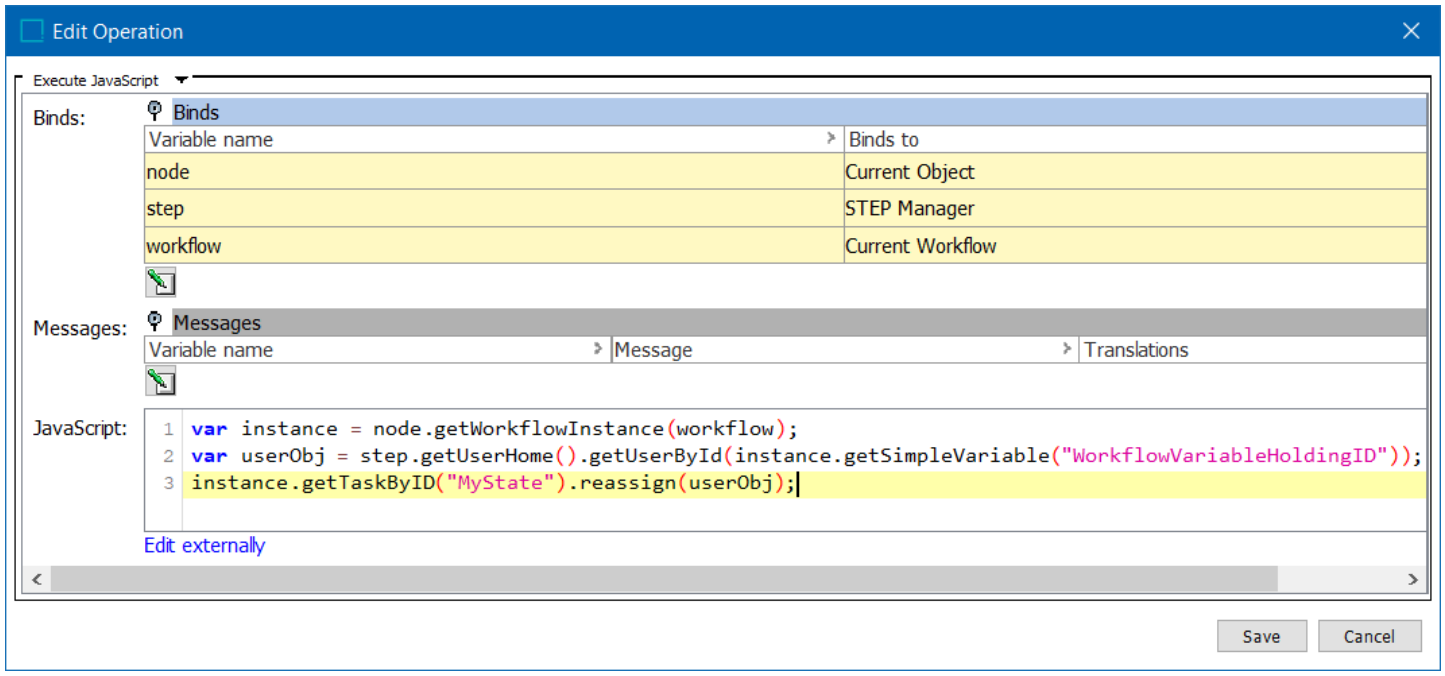
Assigning Tasks Using Business Rules

While the assignee always will be set using the option selected on the State Editor's Assignee tab when entering a state, it is sometimes necessary to overwrite the assignee using a business action configured to run in the State Editor on the On Entry tab.

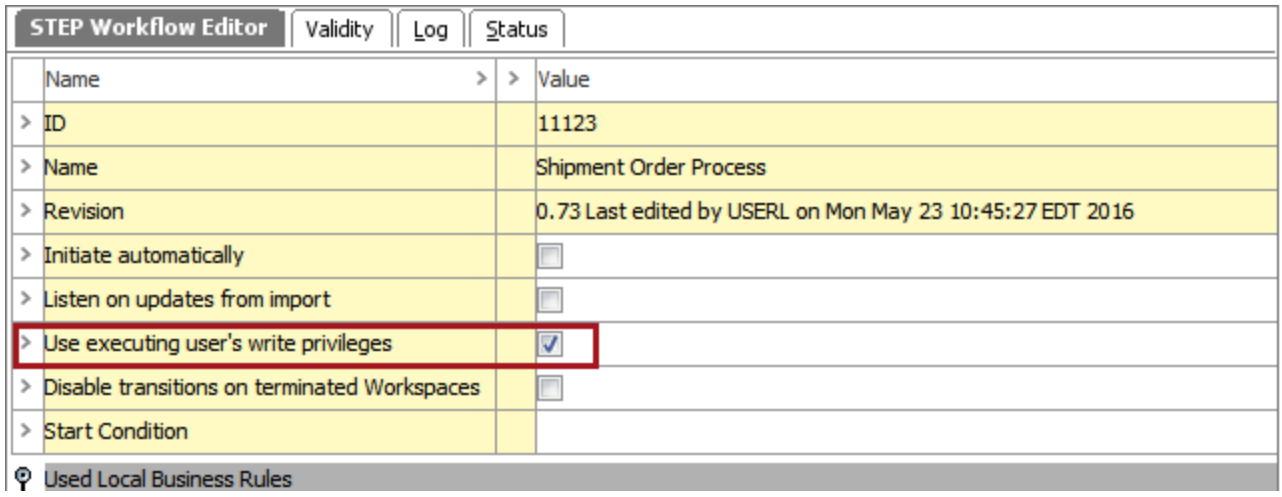


Overwriting the Assignee in Workflows with JavaScript

If the user has the ID of an assignee in a workflow variable, it is of course recommended that the user use the 'Fetch from Variable' option described in the **Workflow State Editor Assignee Tab Option** section of the **Workflows** documentation. If, however, the user wishes to do it with a script On Entry, it can be done with JavaScript, which can be found in the online version of this topic.



A user should have STEP Workflow Administrator privilege to use this 'reassign()' API method, however only if the 'Use executing user's write privilege' option on the Workflow Object editor is selected.

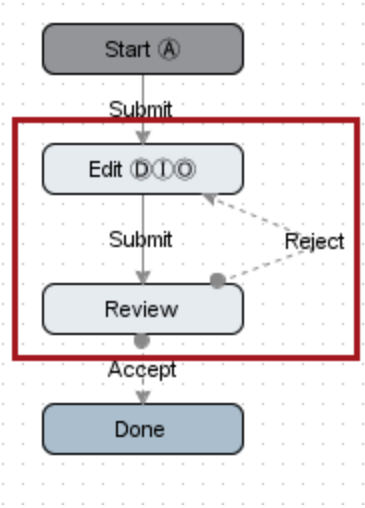


Reassigning to Previous Users in a Workflow

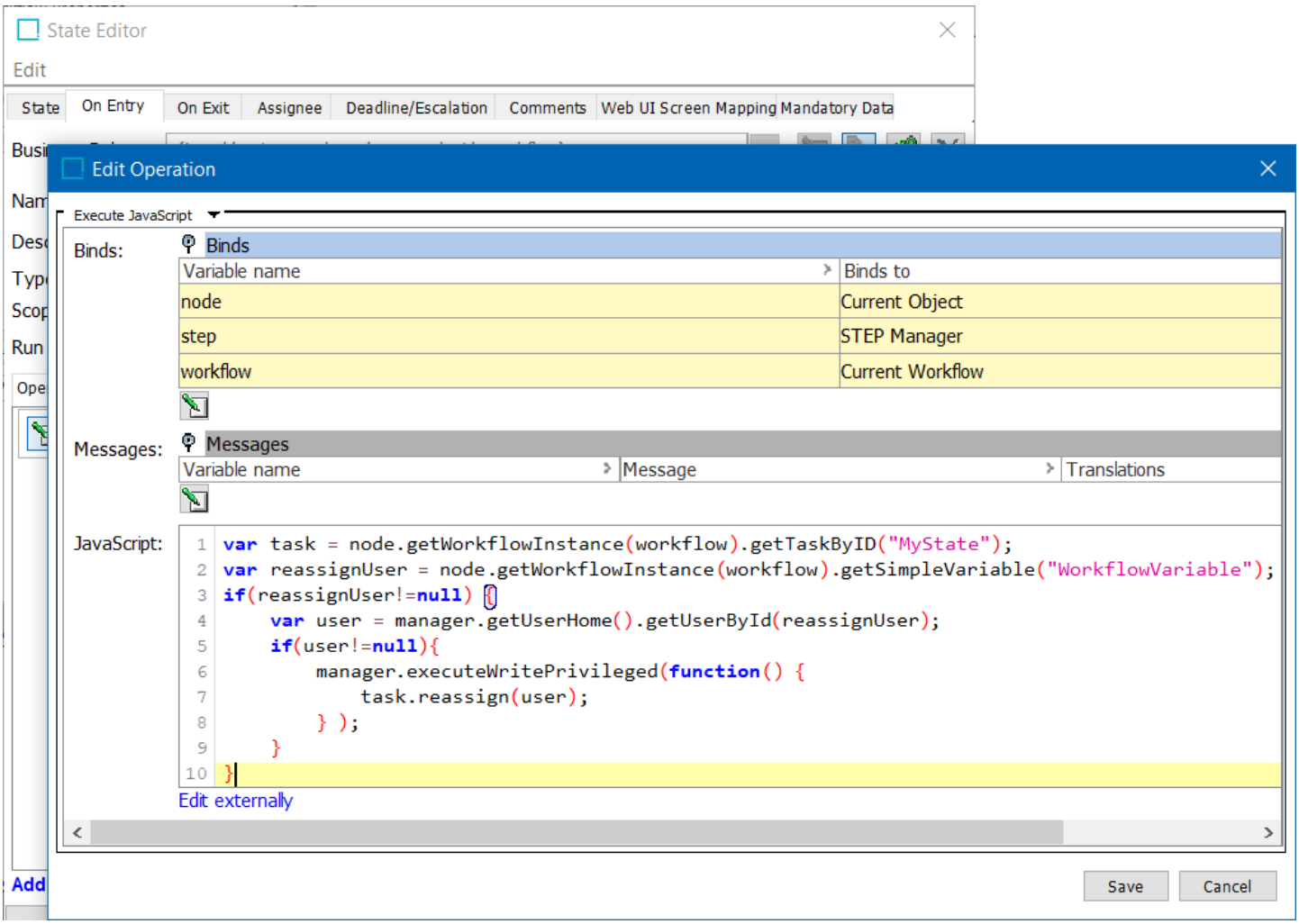
If a task is rejected back to a previous state, and needs to be worked on by the same user who initially worked on it in that particular state, this business rule can be used in combination with On Exit actions to return the correct assignee.

In the example below, a user picks up a task in the Edit sate, completes the assignment and submits it to the Review state. An On Exit business action is in place on the Edit state to store the data of who last worked on the task in a workflow variable. For more about On Exit scripts see **Retrieving the ID of an Assignee Completing a Task in a Workflow** in the **Workflows** documentation.

Once in the Review state, a different user looks at the task and does not think it is ready to move forward. They sends it back to the Edit state to be re-evaluated by the user who originally submitted it.



To do this On Entry action of overriding the workflow default assignee back to the user who originally worked on it in the Edit state, the following script with the corresponding binds need to be put in place:



For the JavaScript example, see the online version of this topic.

Retrieving the ID of an Assignee Completing a Task in Workflows

In a workflow, it is common to have more than one user interact with different tasks in a particular state. As a task progresses through the workflow, it is then helpful to know which user last interacted with the task. To help with this, there is an automated process that can be set to retrieve the ID of the last assignee to a task. This way, should there be an error with the task and it needs to be sent back to the previous state, it can also be determined on who the last user was while in that state.

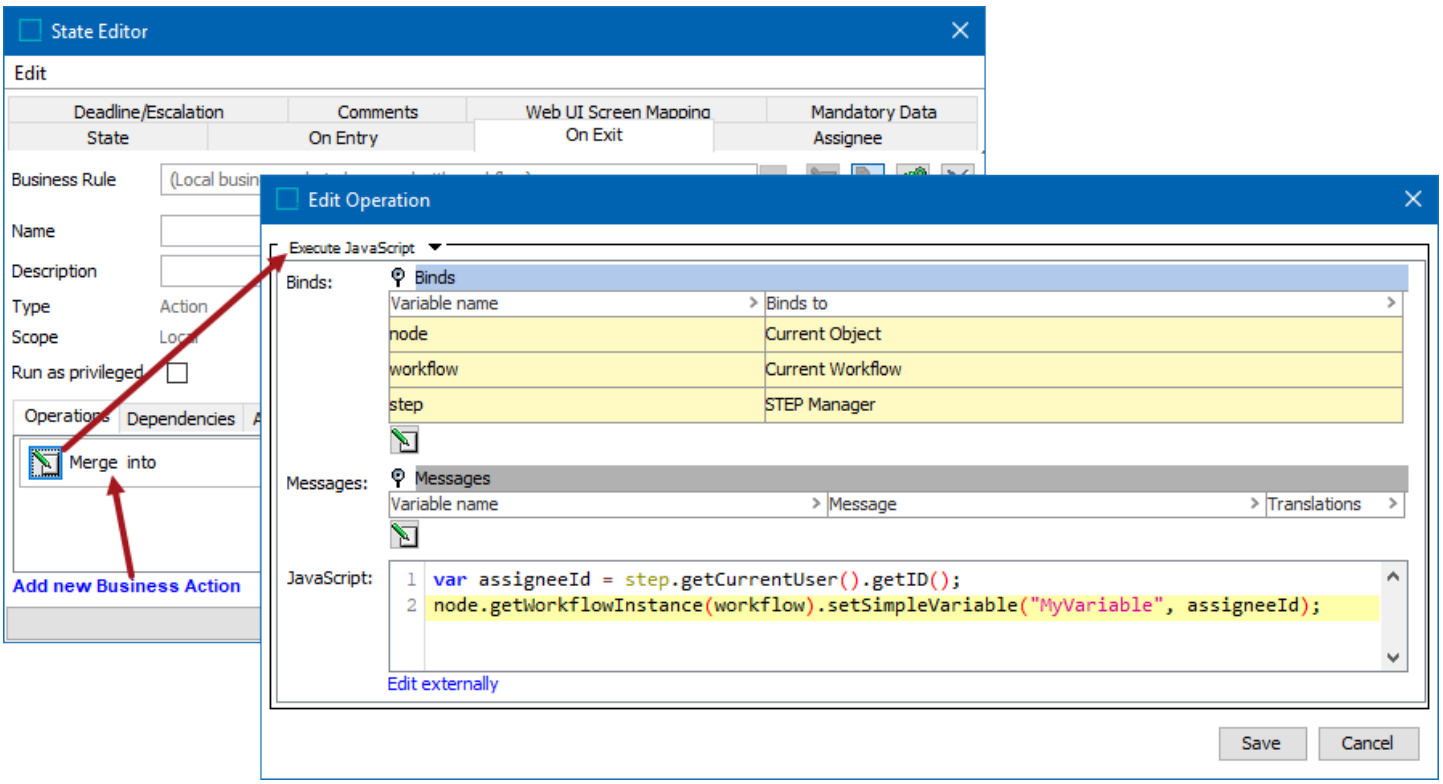
There are two script options that can be used according to user preference. Both are placed on a State Editor on the 'On Exit' tab. Both scripts can also be combined with a condition to guarantee the user ID is returned, if desired.

Using Manager Interface to Return the User ID in a Workflow

In the script below, the ID of the user who performs the transition is obtained via the Manager interface ("step" is a shorthand for getting the Manager). This script retrieves the ID of an assignee and stores it in a workflow variable which is being represented by "WorkflowVariable" in this case.

```
var assigneeId = step.getCurrentUser().getID();

node.getWorkflowInstance(workflow).setSimpleVariable("WorkflowVariable",
assigneeId);
```

For the example below, the workflow variable is called 'MyVariable'.



To implement this, open up the desired workflow, right-click on the desired state and select Edit state > On Exit > Add new Business Action > Merge into > **Execute JavaScript**. Add any necessary binds for the workflow.

One aspect to be aware of is that a user with the 'STEP Workflow Administrator' privilege potentially could have completed the task while the workflow was assigned to a group, resulting in the group ID being returned instead. Use the condition mentioned at the end of this topic if returning the user ID only for the workflow variable is a concern.

Using Task Interface to Return the User ID in a Workflow

Alternately, the ID of the Assignee can also be obtained via the Task interface. This too is implemented by opening up the desired workflow, right-clicking on the desired state, and selecting Edit state > On Exit > Add new Business Action > Merge into > **Execute JavaScript**. Again, replace "WorkflowVariable" with the variable for the workflow, and add any necessary binds for the workflow.

```
var instance = node.getWorkflowInstance(workflow);
var assigneeId = instance.getTaskByID("MyState").getAssignee().getID();
instance.setSimpleVariable("WorkflowVariable", assigneeId);
```

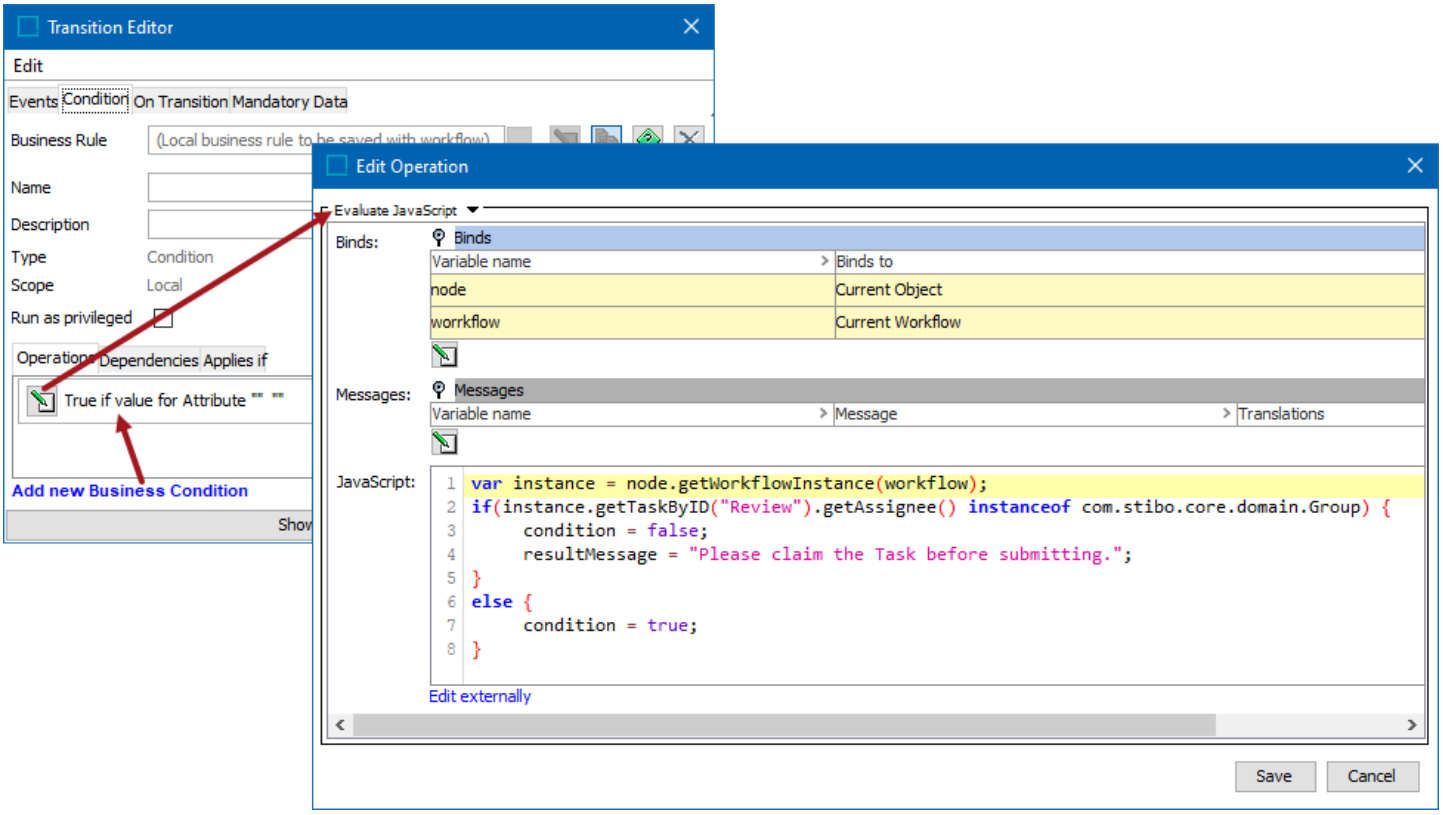
Again, an aspect to be aware of is that a user with the 'STEP Workflow Administrator' privilege potentially could have completed the task while the workflow was assigned to a group. This could result in the ID of a user group being stored in the Workflow Variable, versus the individual user ID.

For this reason, adding a condition on the transitions out of the state in conjunction with the script on the Task interface will guarantee that the ID stored in the variable will be an individual user and not the group ID.

Condition to Aid in returning User ID in Workflows

Below is the condition that can be added to the Transitions Editor on the 'On Transition tab' out of the state that is retrieving the user ID. This condition in conjunction with the script, will make it impossible for a task to exit the state if the assignee is a group ID. To implement the condition, right-click on the transition > Edit Transition > Condition > Add new Business Condition > True if value for Attribute"" "null" > **Execute JavaScript**. Add any binds needed for the workflow.

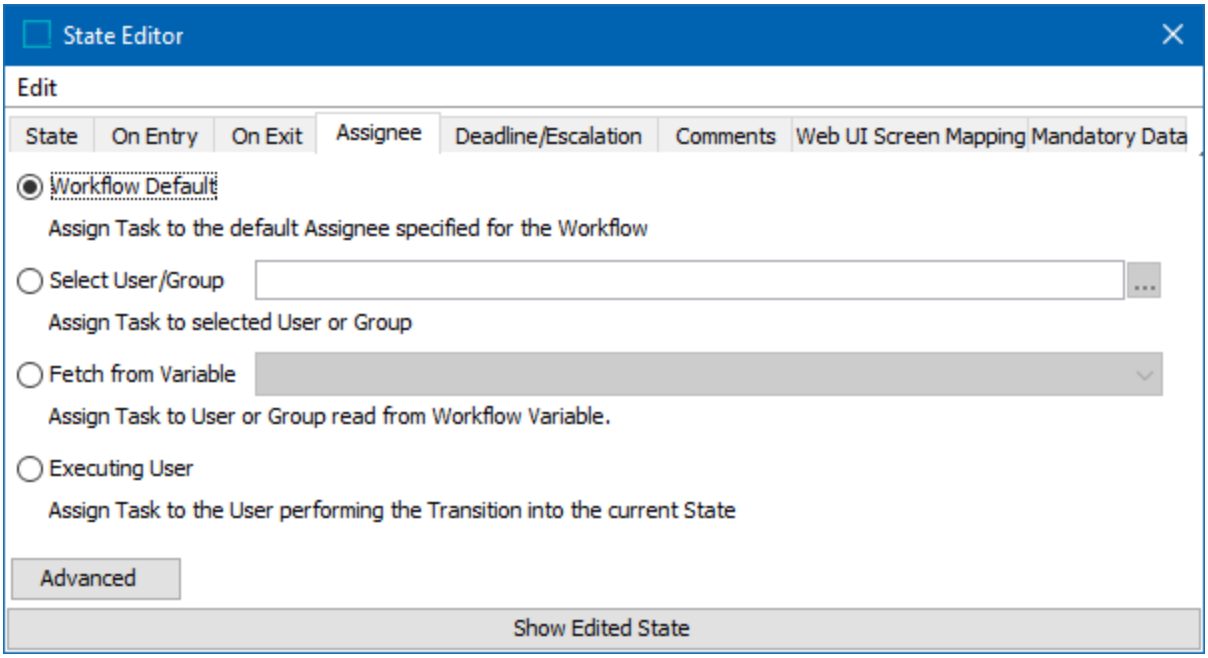
```
var instance = node.getWorkflowInstance(workflow);
if(instance.getTaskByID("MyState").getAssignee() instanceof
com.stibo.core.domain.Group) {
        condition = false;
        resultMessage = "Please claim the Task before submitting.";
}
else {
        condition = true;
}
```



If a user wishes to send a task back to a previous state and re-assign it to the user who worked on the task in that state, see **Assigning Tasks Using Business Rules** topic in the **Workflows** documentation.

Workflow State Editor Assignee Tab Options

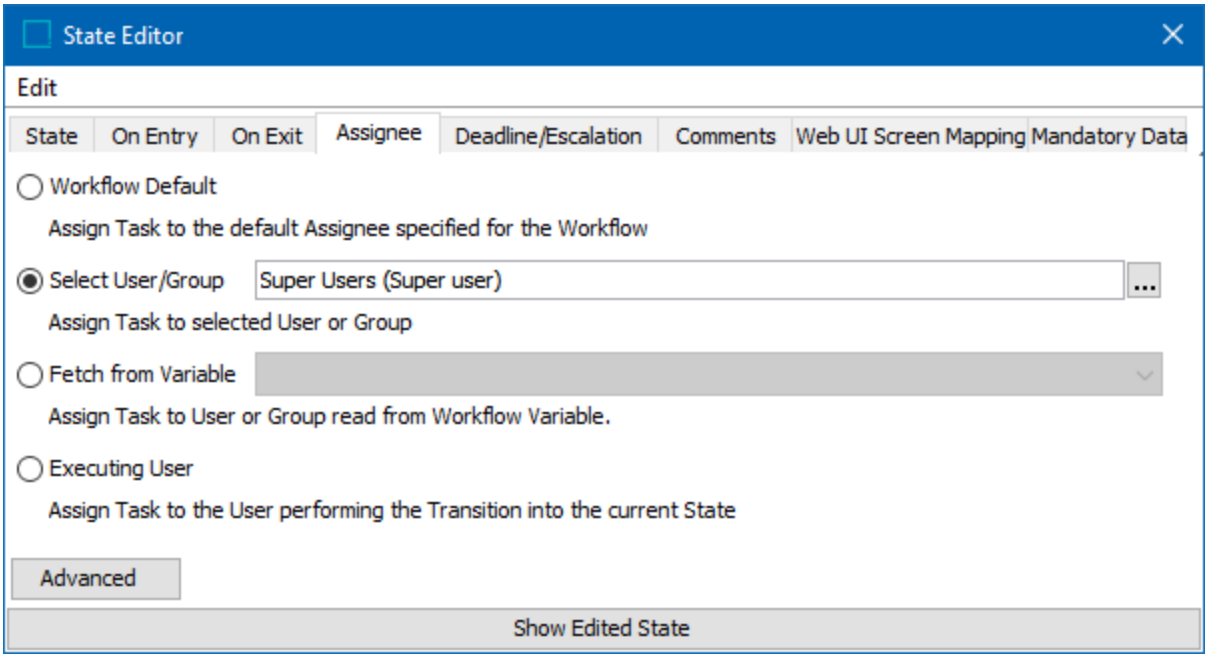
Every state in a workflow needs an assignee in order for the workflow to work. An assignee can be set for any state by opening the State Editor and navigating to the Assignee tab. On the State Editor's Assignee tab the following options for setting the Assignee are available:



Workflow Default

If this option is selected, the state will get the assignee specified as the default for the entire workflow.

The original Workflow Default for the entire workflow is specified by right-clicking the canvas and selecting 'Edit Workflow', by double-clicking the canvas, or by double clicking the header bar showing the Workflow ID/Name. As the user will notice, in the dialog that appears on the Assignee tab, it is not possible to select the 'Workflow Default' option. This is because a user or user group needs to selected in order for there to be a default option for any states in the workflow.



While the user does not necessarily have to specify a default assignee for the workflow, if any of the states are configured to use it, the workflow will not save until a Workflow Default assignee is established.

Select User / Group

Use this option to pick a specific user or user group to function as the assignee.

Fetch from Variable

Using this option it is possible to set the assignee based on the value of a Workflow Variable. When an object is transitioned into a state with this setting, the variable value must hold either a STEP URL for a user or a user group, or the ID of a user or user group (If there is a user and a group with the same ID, the user will be selected).

If STEP cannot find a valid assignee from the variable value, the transition into the state cannot be performed and the user will get an error. As the 'Fetch from Variable' option requires a Workflow Variable to be set before entering, it is not available when setting the workflow default and neither is it available for Initial states not inside clusters.

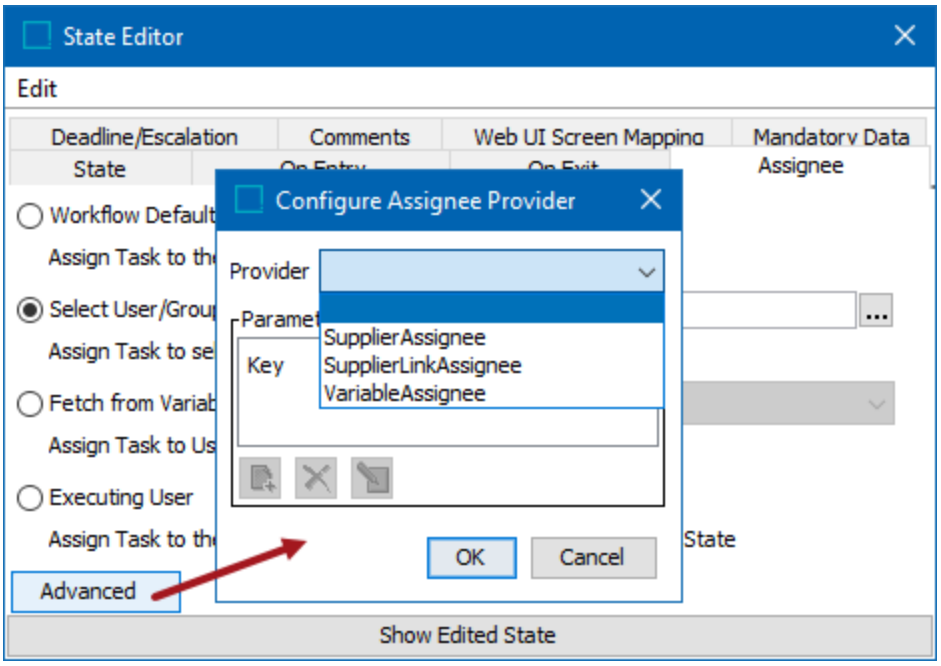
The variable to read the assignee from could be populated by the 'WorkflowVariableUserList' view component (see the **Views and Mappings for Workflows** section of the **Workflows** documentation) or could be set in a business action. An example could be when leaving a state, the assignee for that state is stored. See **Retrieving the ID of an Assignee Completing a Task in a Workflow** in the **Workflows** documentation for an example.

Executing User

Using this option will set the assignee to be the user who performs the transition into the state. Typical use case is when the individual who created the workflow wants the user who starts a task in the workflow to become the assignee for the first state. Beware of the fact that if the option is selected for the entire workflow, the user will be resolved each time a state configured to use the Workflow Default is entered. That is, for each state the assignee will be set to the user performing the transition into the state, not the user who originally started the workflow.

Advanced Options

Via the Advanced button on the Assignee tab it is possible to let the assignee be set by an assignee Provider plugin. These plugins can be customer specific but three are present in the STEP core:



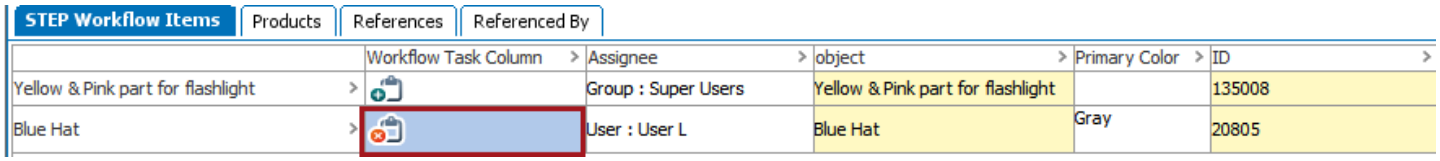
Component	Returned Value
SupplierLinkAssignee	Provider plugin for getting a supplier group assignee via a classification link. Only works for products.
SupplierAssignee	Legacy version SupplierLinkAssignee.
VariableAssignee	Legacy version of the 'Fetch from Variable' option described above.

Note: The assignee provider plugins generally require that one of the above three default options is selected. This selection will then be used as a fallback if the plugin fails in delivering a valid assignee.

Claiming and Releasing Tasks in Workflows

As noted in the **Configuring Views for Workflows** section of the **Workflows** documentation, when using the WorkflowTaskColumn View Component, it is possible for users to claim and release tasks directly from the STEP Workflow Items editor or the object's Tasks tab. This is important because it allows users to 'check out' a task to work on, thus preventing other users from taking the task and working on it as well.

For example, when a user clicks the icon in the column labeled WorkflowTaskColumn on a task assigned to a group, in doing so the user has now assigned the task to themselves and opens it for editing.



This implicitly makes the task invisible to the other users in the group, assuming that they do not have the "STEP Workflow Administrator" privilege.

For normal users that do not have a special 'STEP Workflow Administrator' privilege, the following apply:

- The users can only see tasks assigned to them or a user group they are a member of
- The users can only complete tasks assigned directly to them
- The users can only edit workflow specific data for tasks assigned directly to them

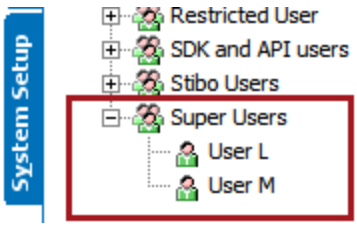
It is important to notice that an object will not be locked when it is active in a workflow. That is, if a user normally is privileged to edit an object, the user can still do this regardless of how many different flows the object is active in. Thus, the assignee related restrictions mentioned above only apply to workflow data and functionality.

When a task is claimed, it is possible to release it again, assigning it back to the latest 'Group Assignee'. If the task has never been assigned to a group, the task is assigned to the default assignee for the state.

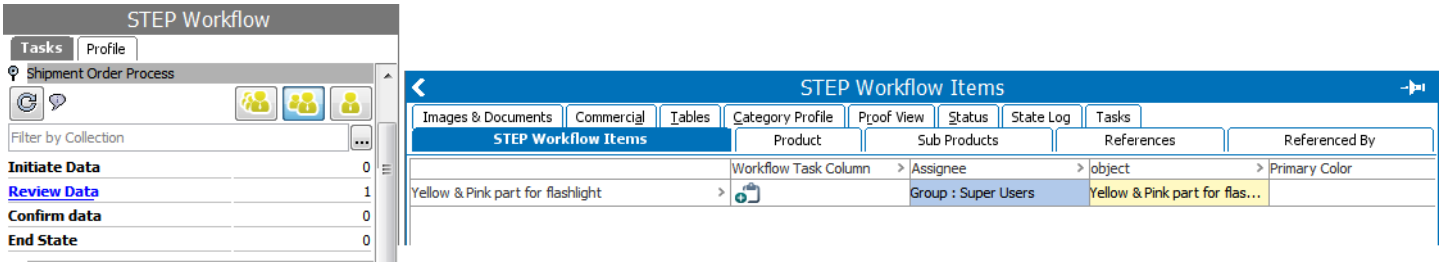
Both the claim and release operations are also available through the API Task interface (Task.claim() and Task.unclaim()).

Claiming a Task in a Workflow

In the following user group, there are two users, User L and User M.

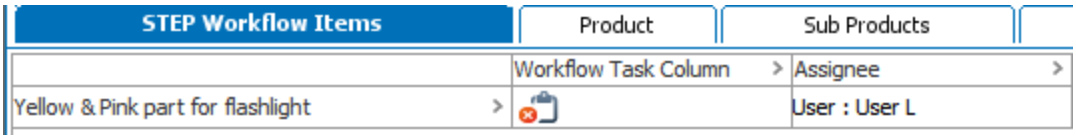


User L signs in, opens the workflow, and navigates to the state where he / she can claim tasks. In this case it is the Review Data state.



Note: Should a task be initiated into a workflow by a user, the task will be assigned to the User Group in which the current user resides.

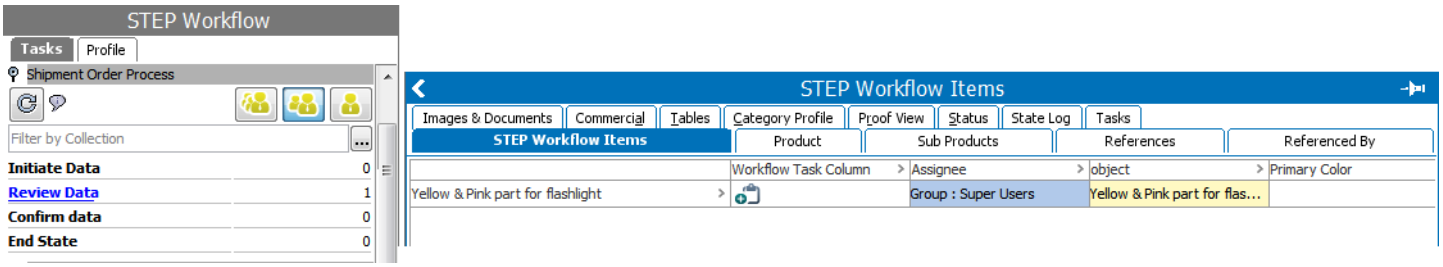
For User L to claim the state, they click on the Claim icon, , claim the task, , and the assignee automatically changes to the user who has signed in, in this case User L.



Once a task has been claimed, it is no longer visible to other users in the same user group.

Releasing a Task in a Workflow

To release a task, click on the Release icon, , and the Claim icon, , reappears. The Assignee column changes back to the user group the user is a part of, and another individual in the same user group can now see the task and claim it if needed.



Workflow Re-assignments

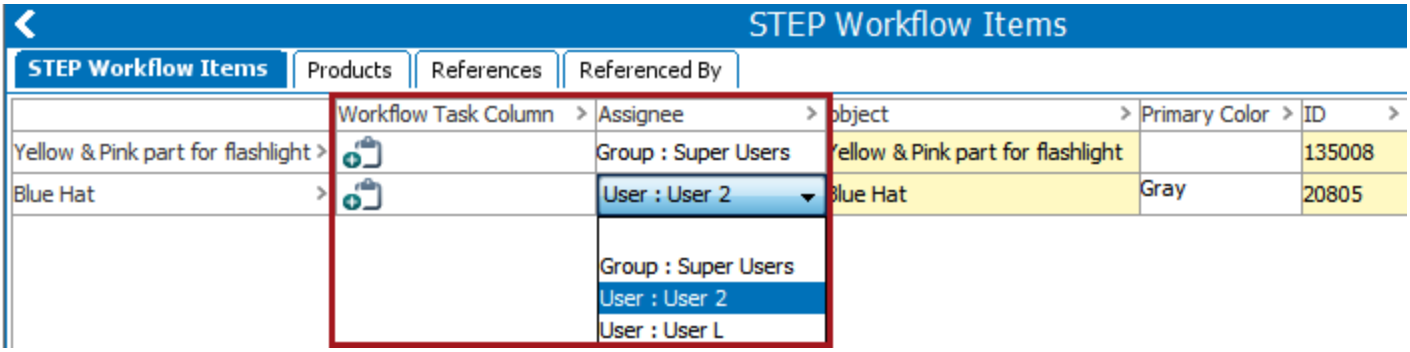
Sometimes it is necessary for an individual user or user group to be assigned a task in a workflow to complete before the task can move on to the next state. For example, a task may be assigned a 24 hour deadline to be picked up by a particular user. If the task is not picked by the user in that time period, then a business rule can be configured in such a way that when the deadline is met, either an email can be sent to a manager to re-assign the task to other user in the user group, or it can be programmed to automatically assign the task to a different user in the group.

Re-assignments might also be helpful if a task needs to be reassigned from one user to another user when a task moves from one state to the next.

There are two ways that a re-assignment of a task can take place. Re-assignments can be performed by a user with the STEP Workflow Administrator privilege either from view and mapping via the 'WorkflowAssignee' Component or via the API Task interface (Task.reassign(assignee)).

Users without the STEP Workflow Administrator privilege will only be able to complete and edit workflow data for tasks assigned directly to them. Thus if a task initially is assigned to a user group but a user needs to work with it, the user must be provided with an option for claiming the tasks, and reassigning it to themselves.

For tasks handled in workbench, a special task view component called 'Workflow Task Column' exists for this purpose. The component lets users claim a task and also assign it back, or release it, to the initial group assignee. More information about this is provided in the **Claiming and Releasing Tasks in Workflows** topic in the **Workflows** documentation.



Discussed below is the setup for STEP Workflow Administrators to perform when configuring re-assignment via the views and mappings option.

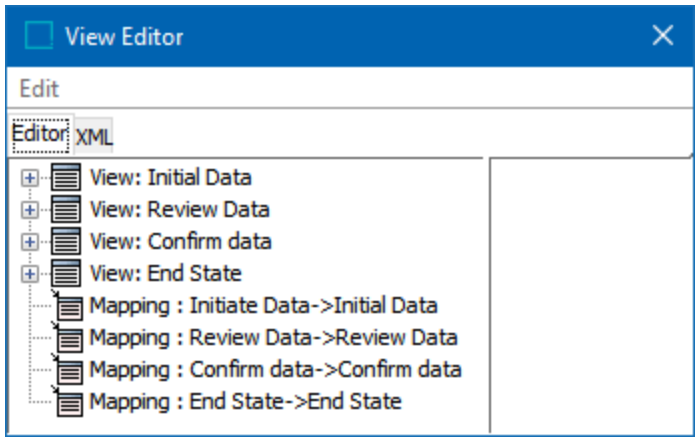
For more on API task re-assignment see the STEP API documentation, accessed by clicking the **STEP API Documentation** button on the STEP Start Page.

Administrator Privilege Setup for Workflow Assignee

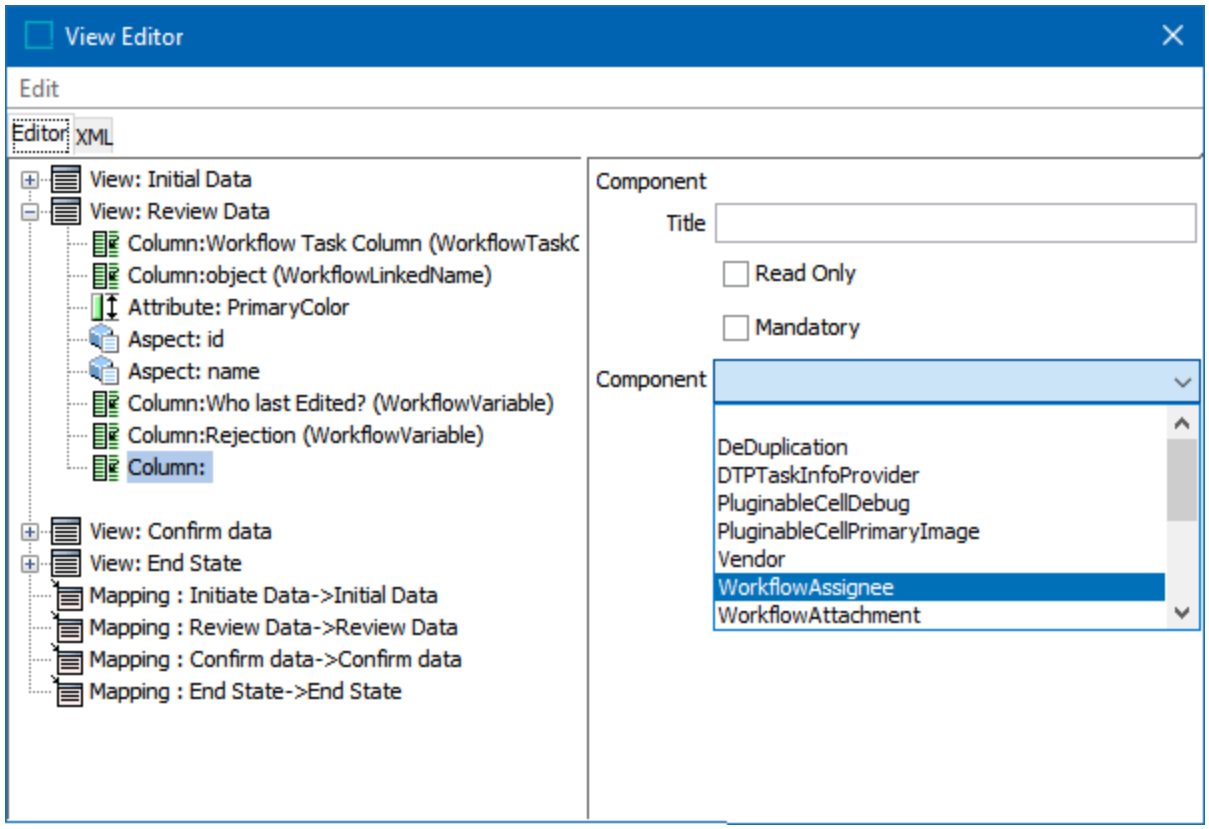
For a user with STEP Workflow Administrator privilege to assign the tasks to different users in a user group, the component 'Workflow Assignee' should be added in the view and mapped to the required workflow state.

For more on views and mapping see **Views and Mappings for Workflows** in the **Workflows** documentation.

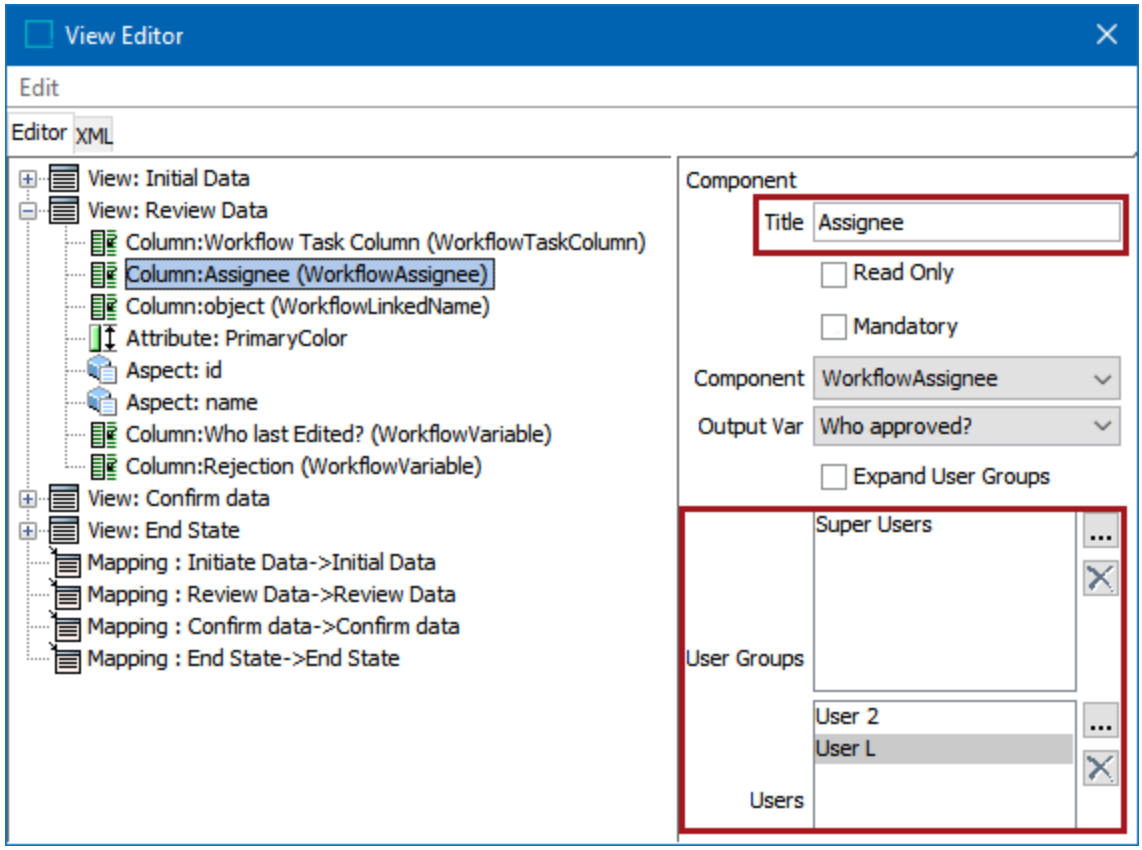
1. Navigate to System Setup and select the desired workflow. Right-click the workflow and select Edit STEP Workflow > **Edit Workbench Views and Mappings**.



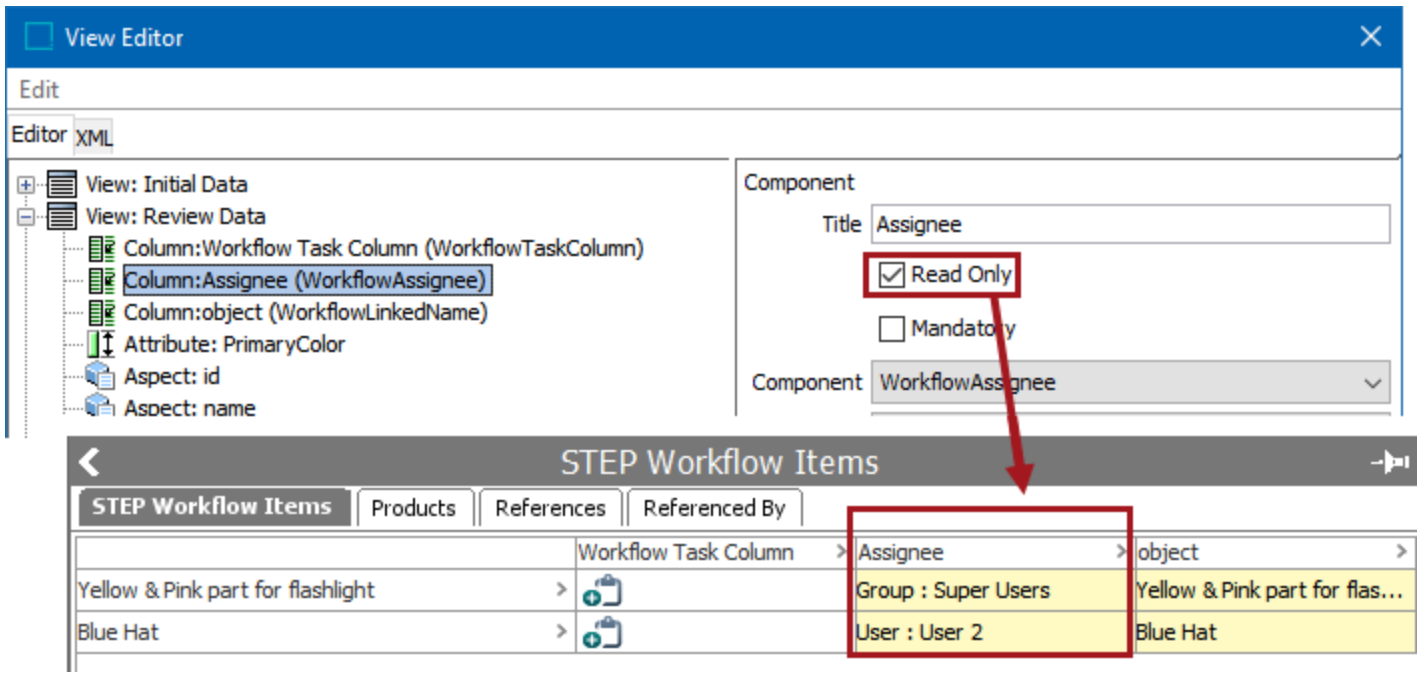
2. Right-click the view that needs assignees and select Add Column > Add Component > **WorkflowAssignee**.



3. Add any user groups and Users to the component as needed, and title appropriately.



It is important to note that if the option 'Read Only' is checked then the Workflow Assignee column will be un-editable, and the user with STEP Workflow Administrator privilege will no longer be allowed to assign the task to other users.

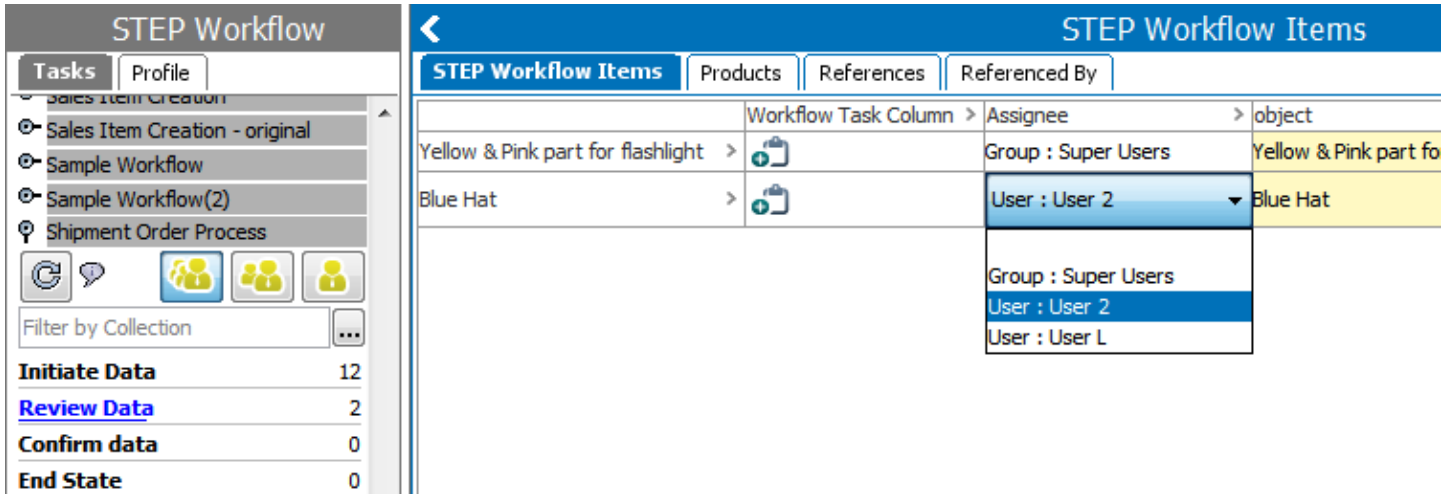


Re-assignment Procedures for STEP Workflow Administrators

To assign a particular task to a user, the user with STEP Workflow Administrator privilege can do so manually on the STEP Workflow tab and the object's Tasks tab, or by creating a business rule. For task reassignment using business rules, refer to the document **Assigning Tasks Using Business Rules** in the **Workflows** documentation.

STEP Workflow Tab

1. Go to STEP Workflow Tab and select the relevant workflow.
2. Click on the particular state in the workflow, and navigate to the 'Assignee' column in STEP Workflow Items tab.
3. Once on the 'Assignee' column, click on the dropdown so that assigned user will be visible.

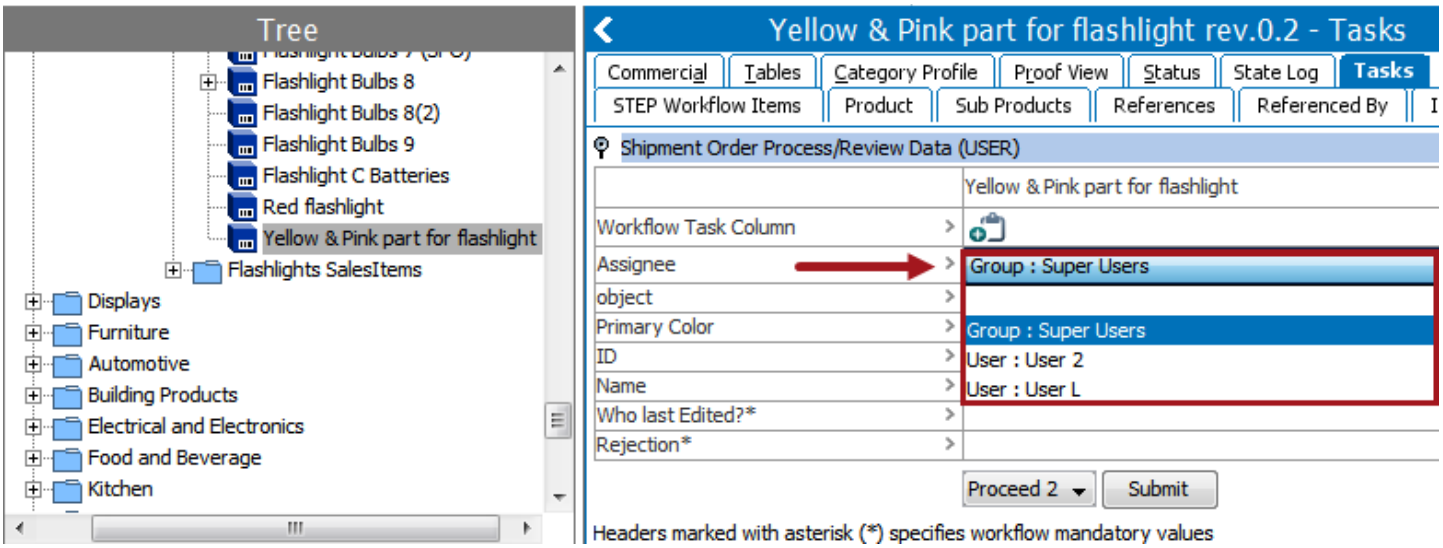


Select the user from the dropdown in parallel with the task to be assigned.

Object's Task Tab

Similarly to how an assignment to a user is performed on the STEP Workflow tab, a task can also be assigned on an object's Tasks tab.

1. To do so, click on the object and navigate to the Tasks tab.
2. Once on the tab, locate the workflow the object is in that needs a task assignment. Select the user it should be assigned to from the 'Assignee' column.



Auto-Initiation of Tasks in Workflows

Automatic initiation is used to automatically initiate an object into the desired workflow, otherwise a manual action is required. With auto-initiation, items can be pushed through automatically, making it much simpler and more time effective.

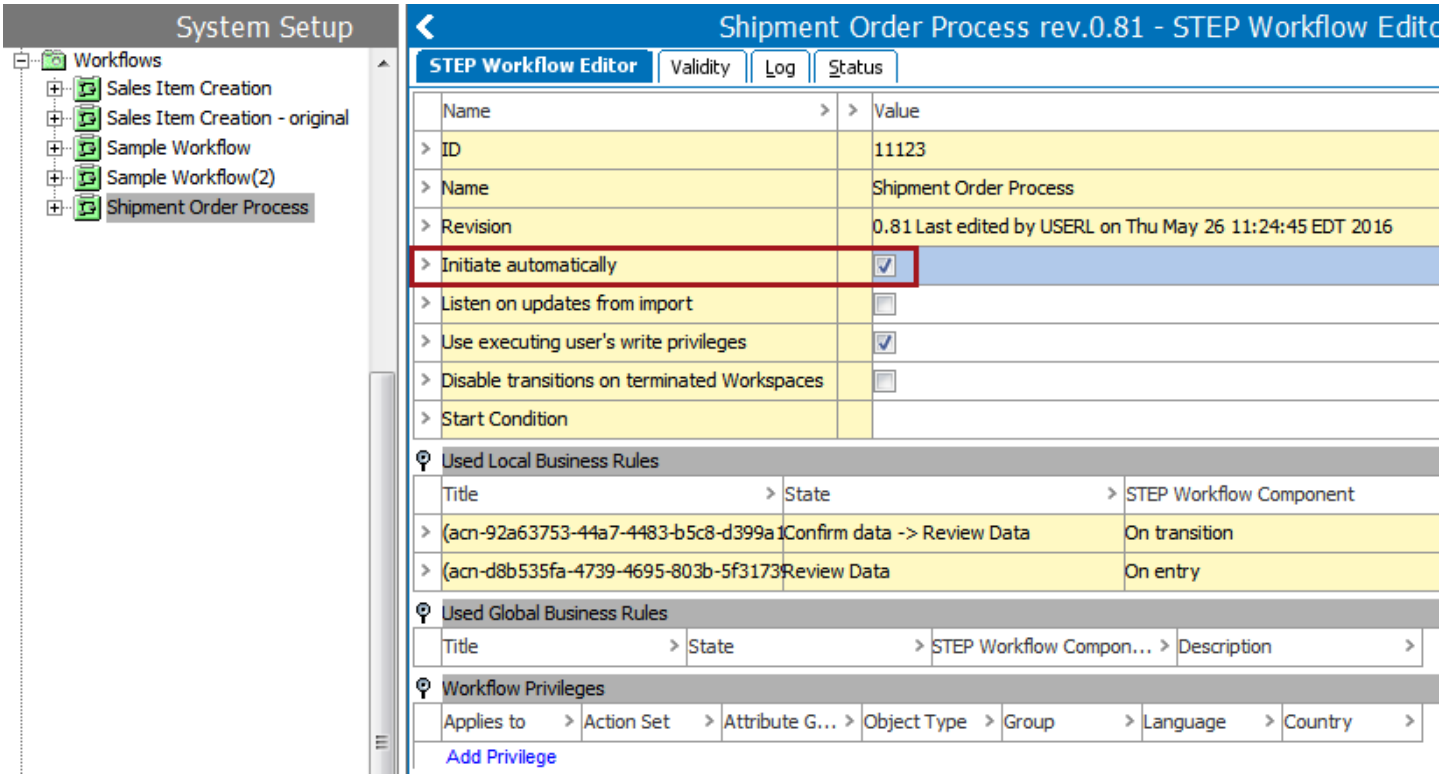
Two common scenarios for auto-initiation are:

1. Once a user creates an object in workbench or Web UI, the object can be initiated automatically in the specific workflow.
2. When new objects are imported, the object can be initiated automatically in the specific workflow.

For example, if a workflow is valid for Products, Classifications, Assets, Entities, Publications, Publication Sections, or Flatplanner Planned Pages, it is possible to have an instance of a given workflow created whenever an object of the valid object type(s) is created. In other words, a workflow is automatically started when the object is created.

Setting up Automatic Initiation of Tasks in Workflows

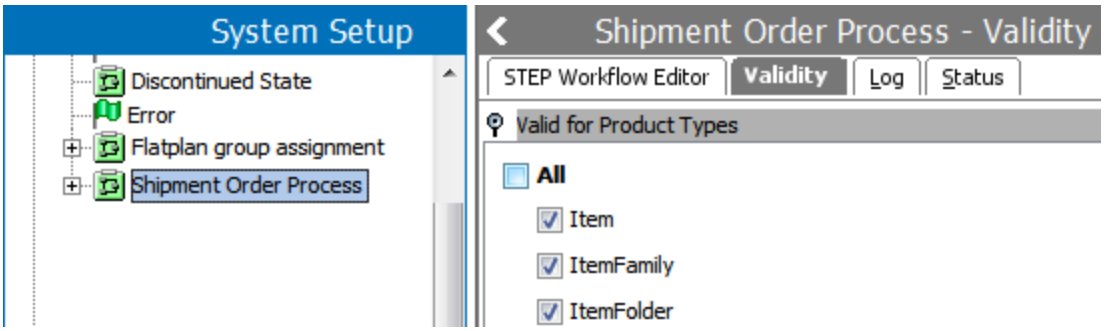
1. To set up auto-initiation, select the System Setup tab and expand the folder that houses workflows.
2. Select the desired workflow (in this example, Shipment Order Process will be used) and check the 'Initiate automatically' checkbox on the STEP Workflows Editor tab.



Whenever an object of a type valid for the workflow is created, a workflow is started. This could happen via workbench, Web UI, web services, or through an import.

Initiating an Object Automatically in Workflows

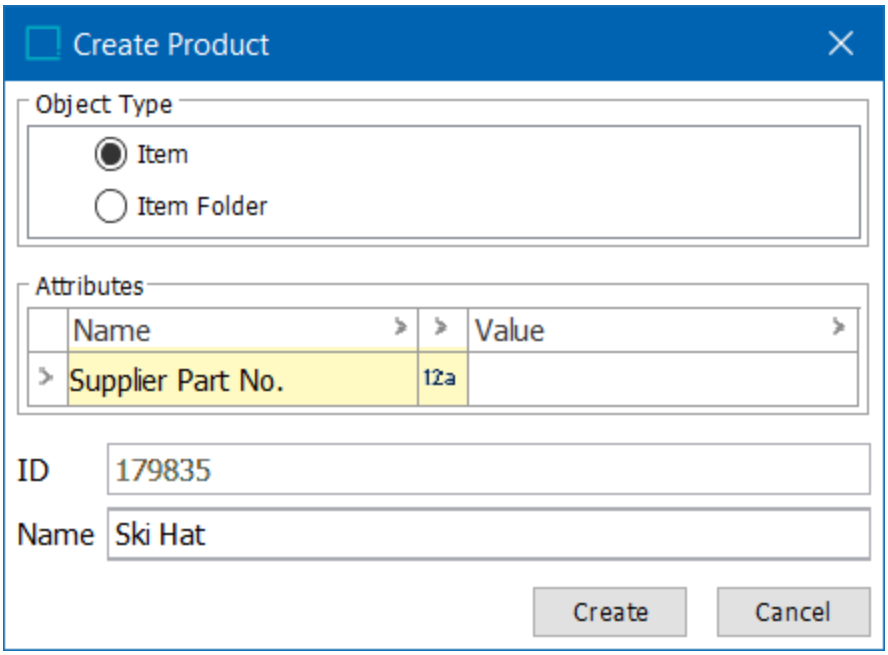
Before creating an object, it is important to know what the desired workflow is valid for, otherwise the object will not be able to automatically initiate. To check the validity type, view the workflow and select the Validity tab. In this example, the Shipment Order Process workflow is valid for products that have one of the three object types below.



Note: Object types / names will vary according to user setup.

Create and Initiate from Workbench

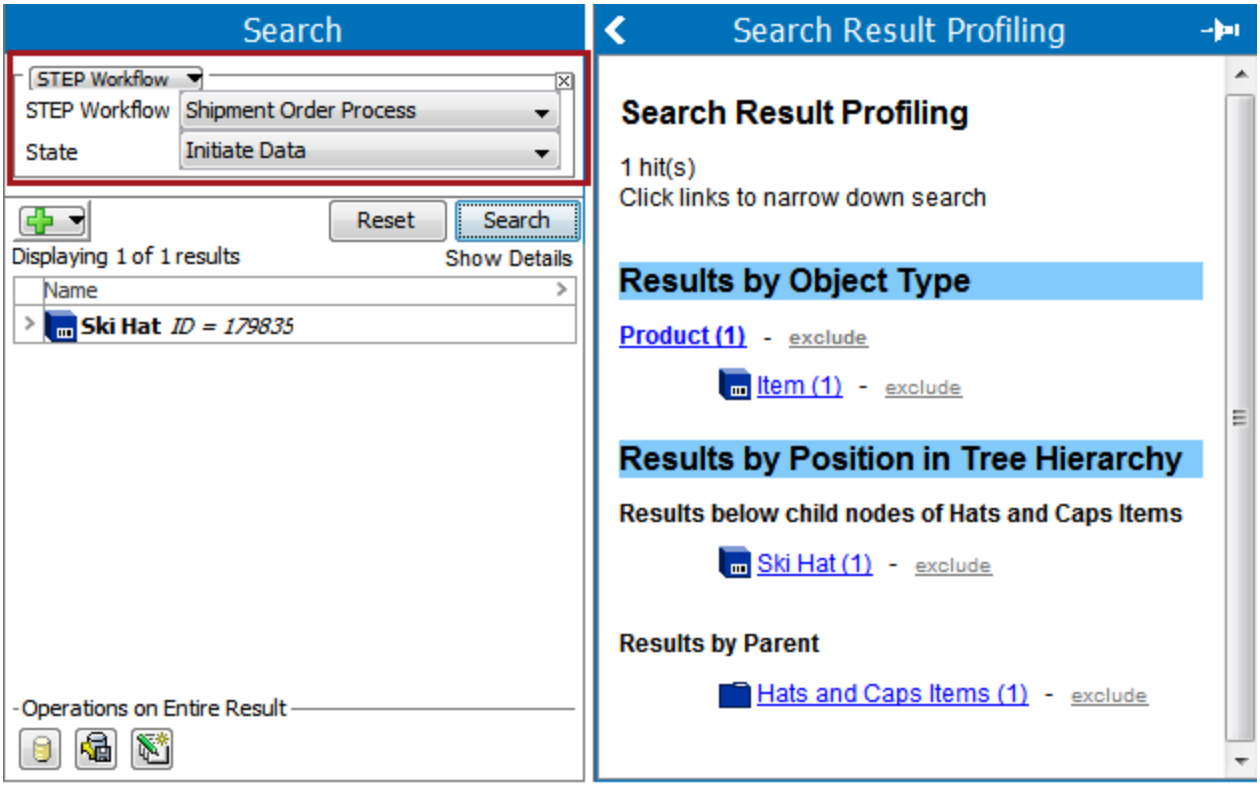
Once the validity type is verified, a product can be created in the workbench by entering an ID and name, as well as any other required data. The following image shows that a Supplier Part Number is required to create an item.



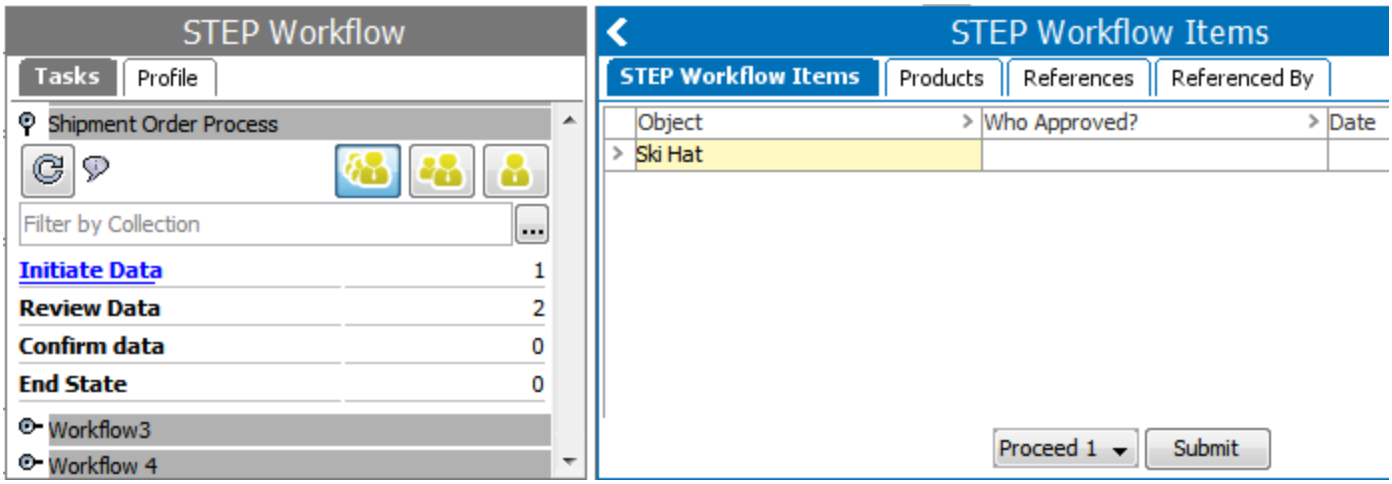
For more information on creating new objects see the **Creating Objects in Tree** topic in the **User Guide / Getting Stared** documentation.

There are a number of ways to verifying that the product was automatically initiated into the desired workflow. A few are:

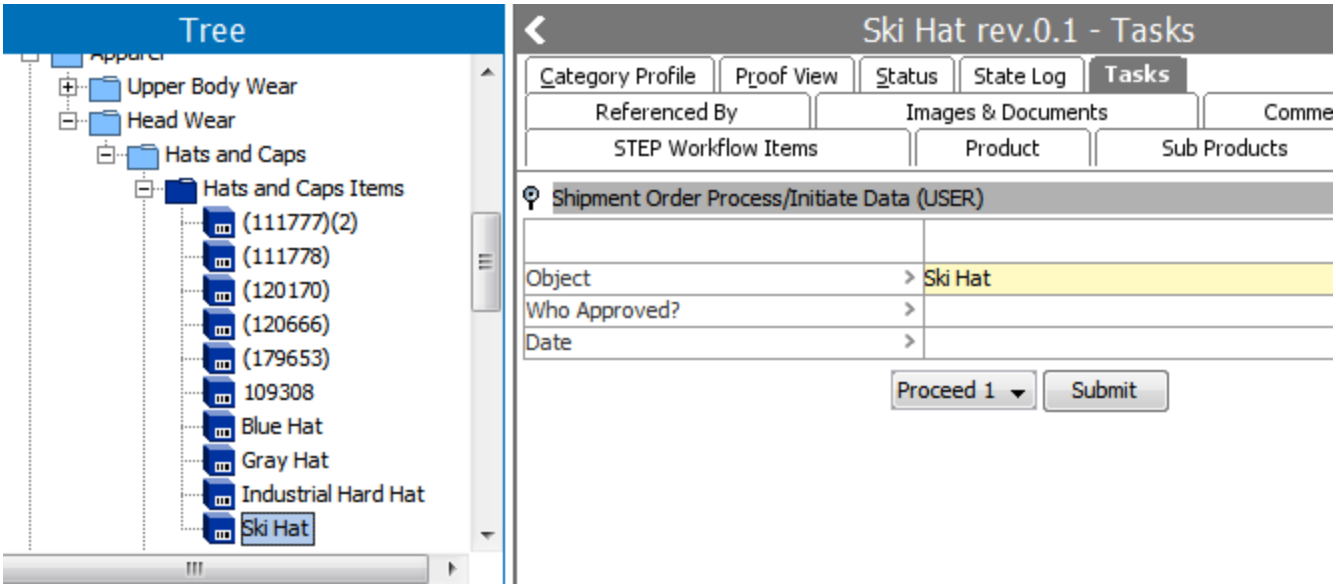
- On the Search tab, search the workflow to verify the object is displayed as a result.



- Navigate to the STEP Workflow tab, open the workflow, and verify that the object appears in the first state in STEP Workflows Items tab.



- Navigate to the item's Task tab and verify that the object appears in the desired workflow flipper.



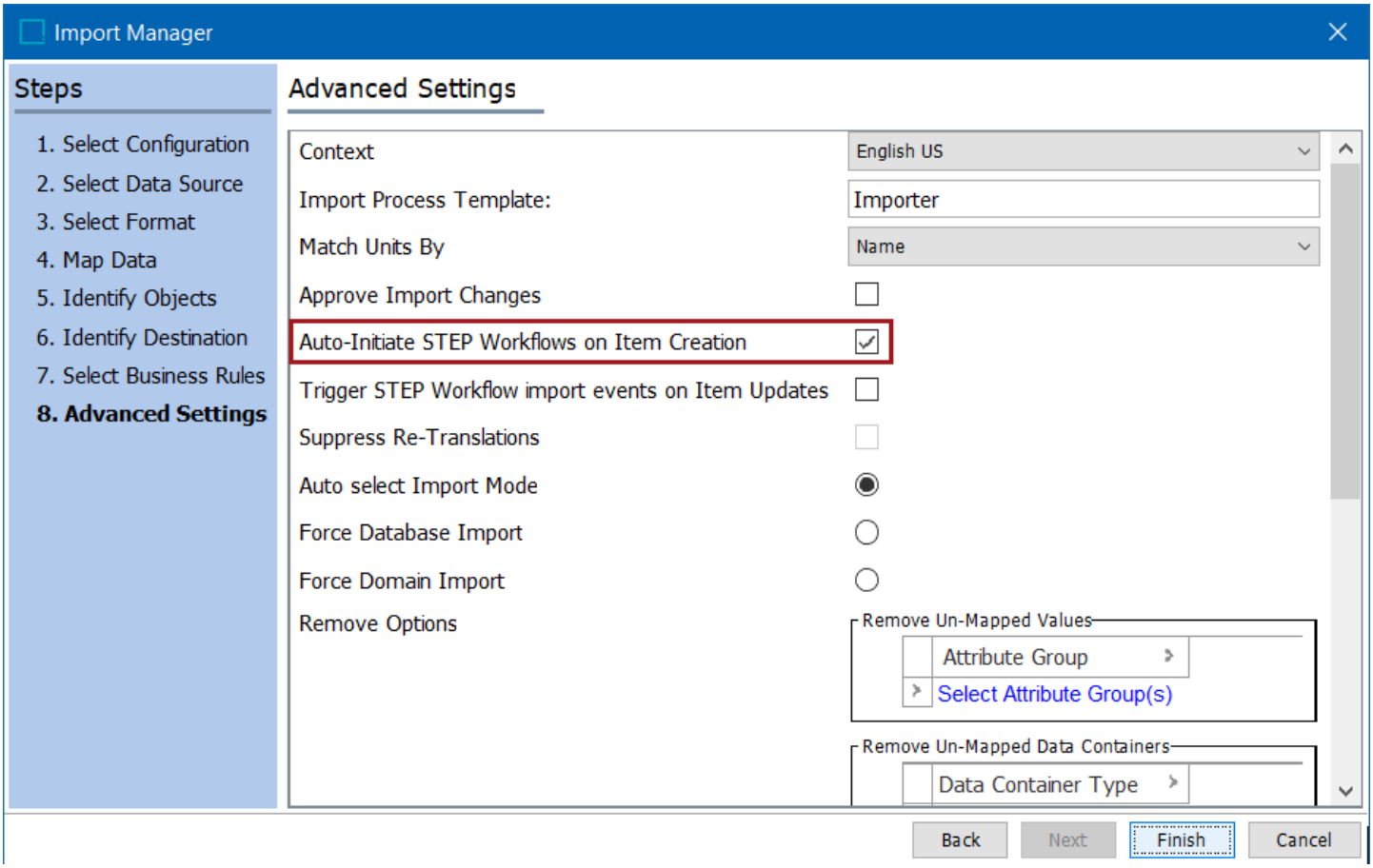
Note: If an object in the Recycle Bin (and thus taken out of any workflow) is revived, it will be re-initiated in the workflows with auto-initiation turned on and that match the object's validity.

Create and Initiate from Import

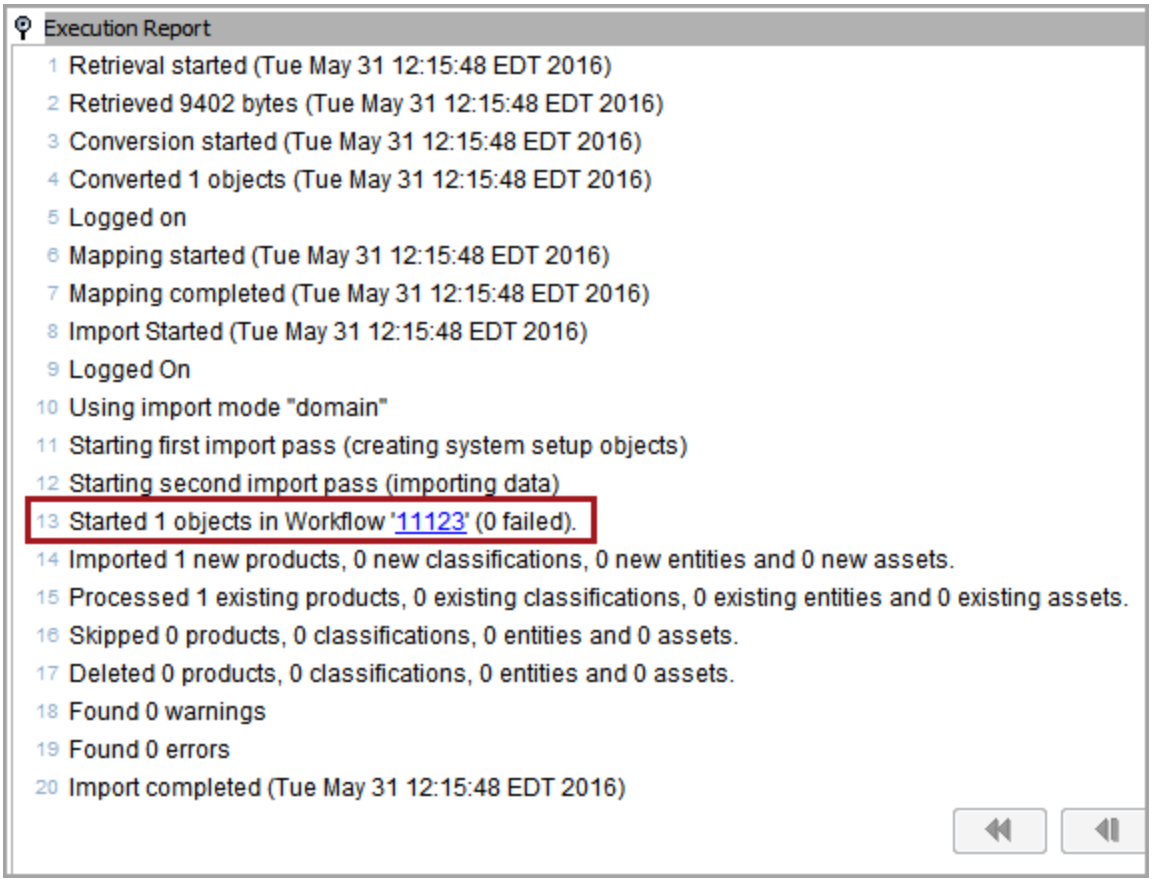
Another way to create and initiate objects into a workflow automatically is to do so via an import, where product information, such as that below, is uploaded to the STEP Workbench.

	A	B	C	D
1	<ID>	<Name>	<Parent ID>	<Object Type Name>
2	179835	Ski Hat	20436	Item
3				

To automatically initiate items upon import, in the Import Manager wizard, under 'Advanced Settings' check the **Auto-Initiate STEP Workflow on Item Creation** option.



When the import is run, a background process is created and the user can see the object(s) initiated into the workflow in the background process.



For more details on the Import Manager functionality and the Import Manager wizard, see **Creating a Data Import** in the **Data Exchange** documentation.

Disabling Auto-Initiation

If a user has the 'Disable STEP Workflow Auto-initiation in Imports' privilege for a workflow, he/she is allowed to disable auto-initiation in the Import Manager (column based Imports) or directly in the XML for XML imports (by setting AutoInitiate="N" in the STEPProductInformation tag as shown below).

```
<?xml version="1.0" encoding="utf-8"?>
<STEP-ProductInformation ContextID="Context1" WorkspaceID="Main" AutoInitiate="N">
<Products ID="179859" DefaultParentID="20436" DefaultUserTypeID="Item">
<Product/>
</Products>
</STEP-ProductInformation>
```

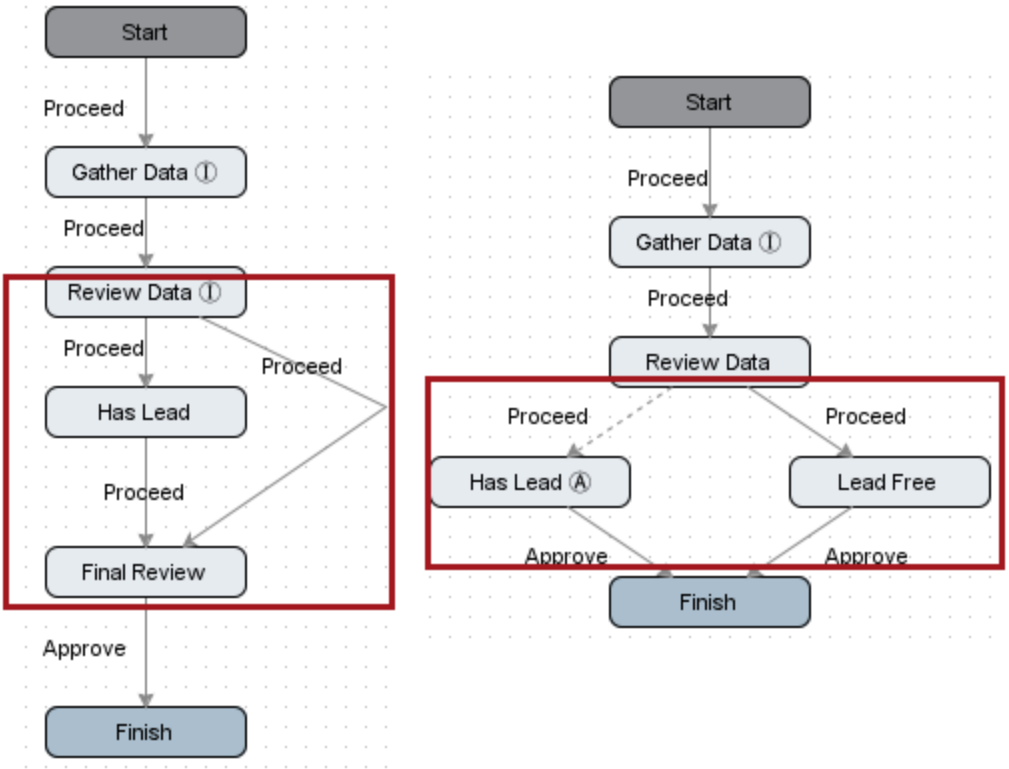
Auto-Routing Tasks in a Workflow

In workflows, there may be cases where the system (via business rules) should make the decision for the path a task takes through the workflow, rather than a user. While there are many use cases for this, a simple scenario is the 'if X, then Y' reasoning. For example, a user may have a workflow set up that evaluates each product that passes through a particular state to see if there is lead in it. If so, special handling instructions need to be applied, whereas this is not required for items that are lead-free. Therefore, tasks are automatically routed to differing states based on the presence / absence of lead, which is determined by an attribute value.

This topic describes the preferred method for basic auto-routing of tasks, and allows for users to submit a task without requiring any knowledge of where it needs to go next, as this is evaluated and determined via a series of conditions on the available transitions. There are additional methods available for auto-routing of tasks, including via use of the Scripting API Task interface and the triggerByID() and triggerLaterByID() methods.

Auto-routing using conditions requires that all transitions from the state have the same event name applied. It is also important to remember that all conditions return either true or false, and that a condition evaluating to true on a transition means that the transition can be taken, while a condition evaluating to false means that the transition cannot be taken.

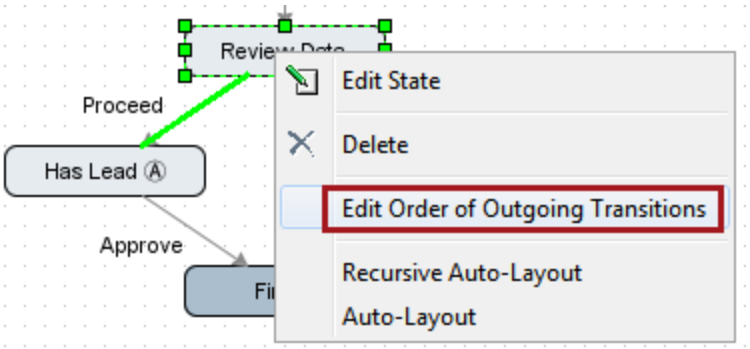
This setup can be used to skip states, as well as to select between one of two states. Therefore, either of the below workflows could utilize it, and in both cases the outcome will be that the task is sent to the Has Lead state from the Review Data state, if needed. Note that in all cases, all transitions out of the Review Data state have the Proceed event. With this setup, the end user will see only a single selection option (Proceed) when submitting the task, allowing the system to make the decision, while keeping things simple for the end user.



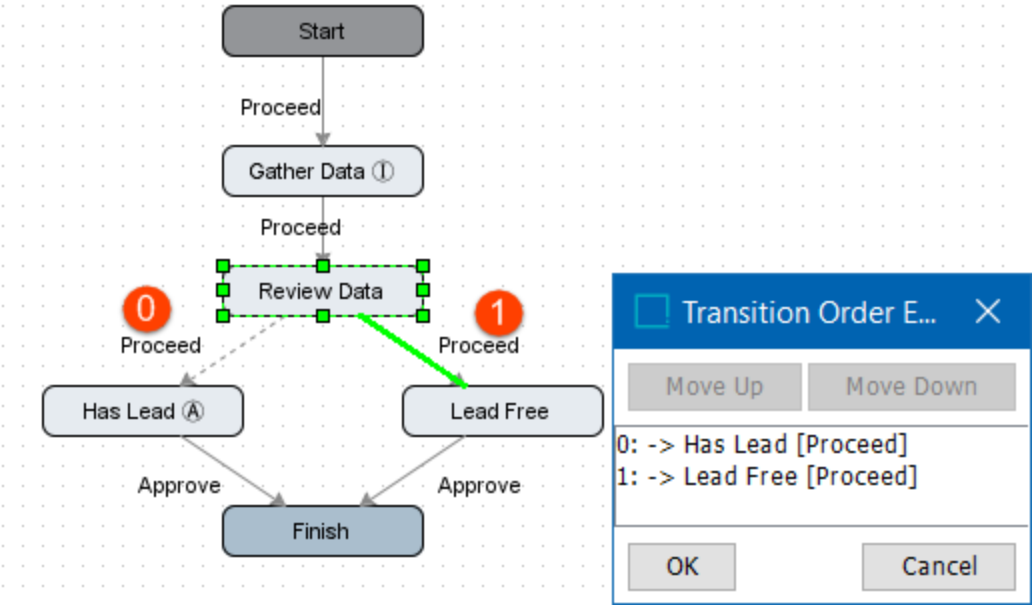
In order to determine where the task should go, an attribute with ID=HasLead is evaluated. If the value is Yes, the task must go to the Has Lead state. If the value is No, the task must take the other transition. We could set up a "Yes" condition and a "No" condition, applying one to each transition out of the state as appropriate, and that would accomplish the end result. However, if an object did not have the HasLead attribute populated, it would not have a viable transition, unless we wrote a condition to account for a null value. Assuming mandatory attribute functionality was applied and we could ensure a value was present, there is however still an easier way to do it using only a single condition, which evaluates to true if HasLead=Yes. By leaving another transition with no conditions applied, and instructing the system to always evaluate the conditioned transition first, all tasks with the Yes value are caught first and appropriately routed, and all tasks with a 'No' or null value take the other (unimpeded) option.

Detailed instructions and additional considerations for this setup are below.

1. Set the order in which outgoing transitions are attempted by right-clicking on the state that has the multiple transitions with the same name coming out of it, and selecting **Edit Order of Outgoing Transitions**.

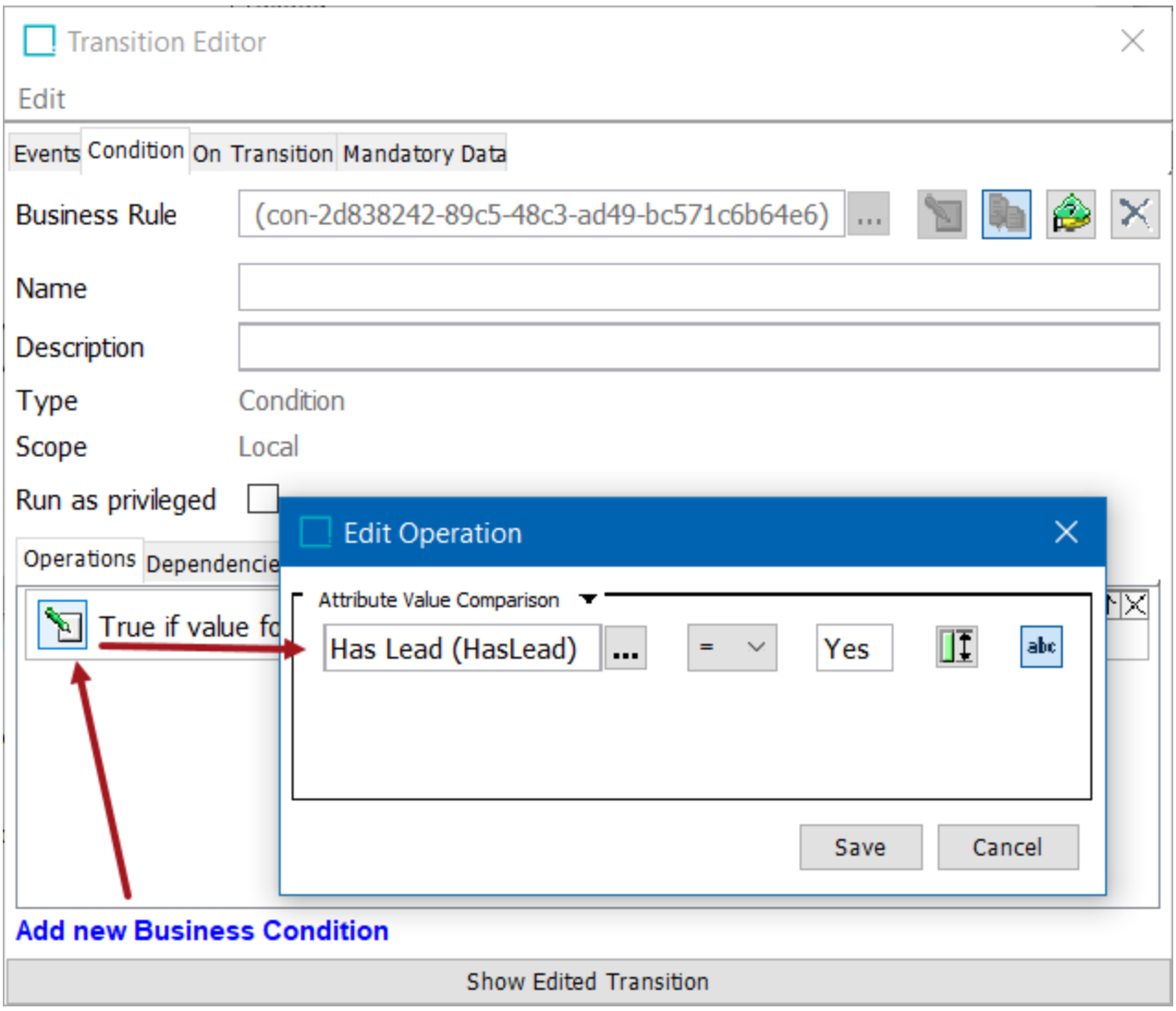


2. In the dialog that appears, select any entry and use the **Move Up** or **Move Down** buttons to set the appropriate order. Note that the example uses only two transitions, but this method works for any number of transitions. The '0' transition is evaluated first, and subsequent transitions are evaluated in sequential order. The system will send the object via the first transition it finds that evaluates as true.



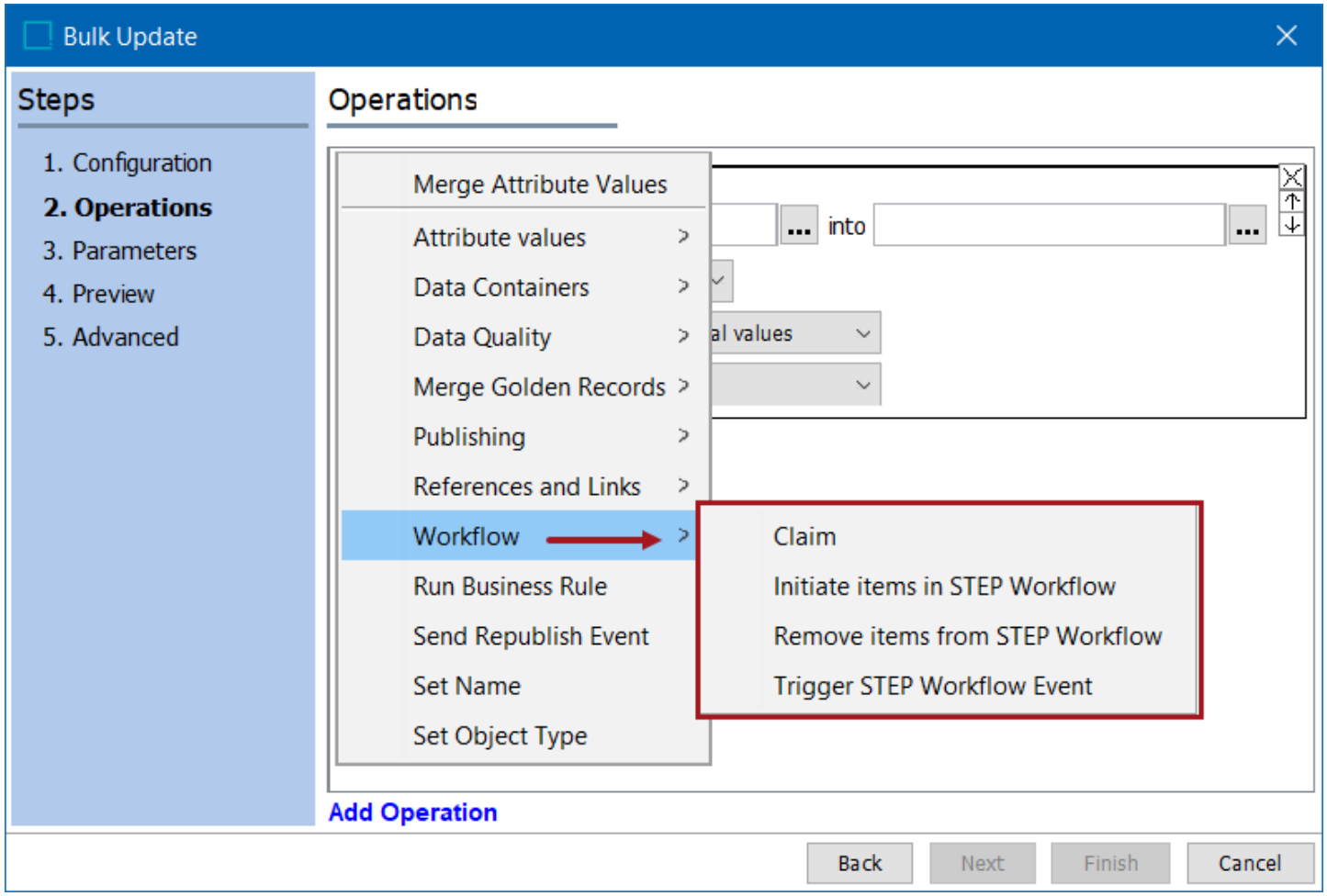
If multiple transitions will exist with conditions applied, care should be taken to ensure that they are sequenced properly, that they do not overlap unless intended (and managed via sequencing), and that all tasks have a way to exit the state. With that in mind, it is generally advised to ensure that conditions are exclusive (e.g., a task is not likely to evaluate true for more than one transition), and that one transition exists unimpeded (e.g., without a condition). However, there are exceptions, so carefully test the design and outcome.

3. Set a condition on the transition to be evaluated first. For the purposes of the example, a basic attribute value comparison will be done, but any condition could be used. Double-click on the order zero transition to open the Transition Editor. Select the **Condition** tab and click the **Add new Business Condition** link. Click the **Edit** icon to open the Edit Operation dialog and select **Attribute Value Comparison** from the dropdown. Select the appropriate attribute, operator, and value. Click **Save** to close the Edit Operation dialog, and click the **X** in the upper right corner to close the Transition Editor.


Transition Editor
Edit
Events
Condition
On Transition
Mandatory Data
Business Rule
(con-2d838242-89c5-48c3-ad49-bc571c6b64e6)
Name
Description
Type
Condition
Scope
Local
Run as privileged
Operations
Dependencie
True if value fo
Edit Operation
Attribute Value Comparison
Has Lead (HasLead)
=
Yes
abc
Save
Cancel
Add new Business Condition
Show Edited Transition

Bulk Update Operations for Workflows

The STEP Bulk Update operations is an efficient way of updating multiple objects in a single operation. For workflows it can be used to claim multiple objects in a specific state in a specific workflow, initiate multiple objects into a specific workflow, remove multiple objects from a specific workflow, or trigger a specific event for multiple objects in a specific workflow.



For more on bulk updates see the **Bulk Updates** documentation.

The table below defines how each operation is expected to perform. Additionally, the next proceeding sections of this document will explain how each operation works more precisely.

Operation	Description
Claim	Assigning a workflow task to a specific user
Initiate items in STEP Workflow	Bulk update is performed on the objects to initiate them into a selected Workflow.
Remove items from STEP Workflow	Removes all objects from the selected workflow.
Trigger STEP Workflow Event	Triggers a selected workflow event.

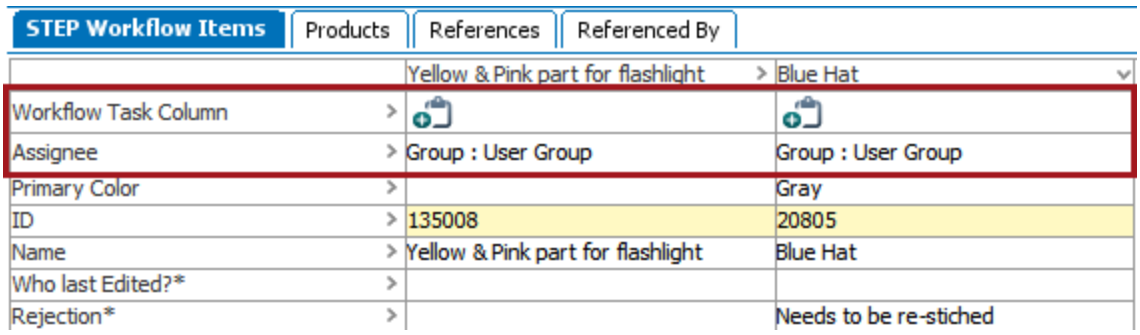
Claiming a Task in a Workflow

The 'Claim' bulk update operation allows the user to claim multiple tasks in a state at one time. For example, if there are 10 tasks in one state, instead of a user having navigate to the STEP Workflow tab, opening up the desired workflow and clicking each individual task using the claim button, , in the STEP Workflow Items tab, a user can claim all the tasks at once using bulk updates.

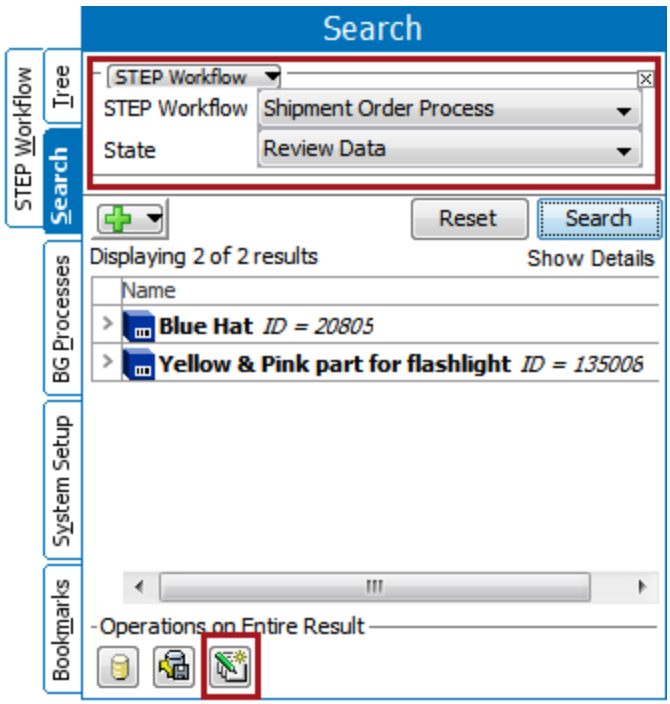
Claiming a Task

To claim tasks, the following steps need to be followed by the user who wishes to claim the tasks. In this case it is User L.

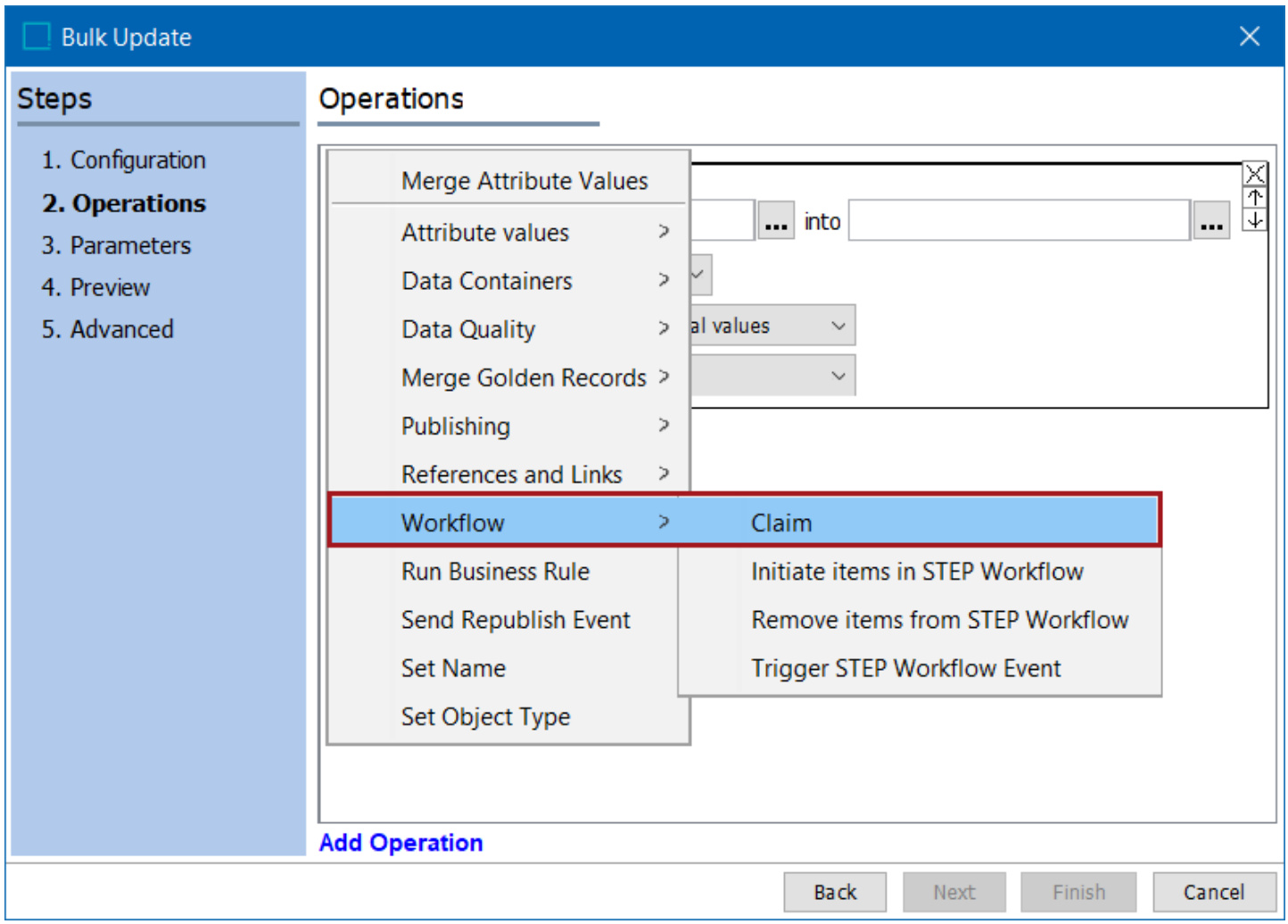
1. It is important to check and make sure that the tasks are initially unassigned to any user in a state.



2. Once verified that they are unclaimed, go to the Search tab, and select STEP Workflow from the dropdown. Next select the desired workflow and state to display the objects present in that particular state, and click the bulk update button, .

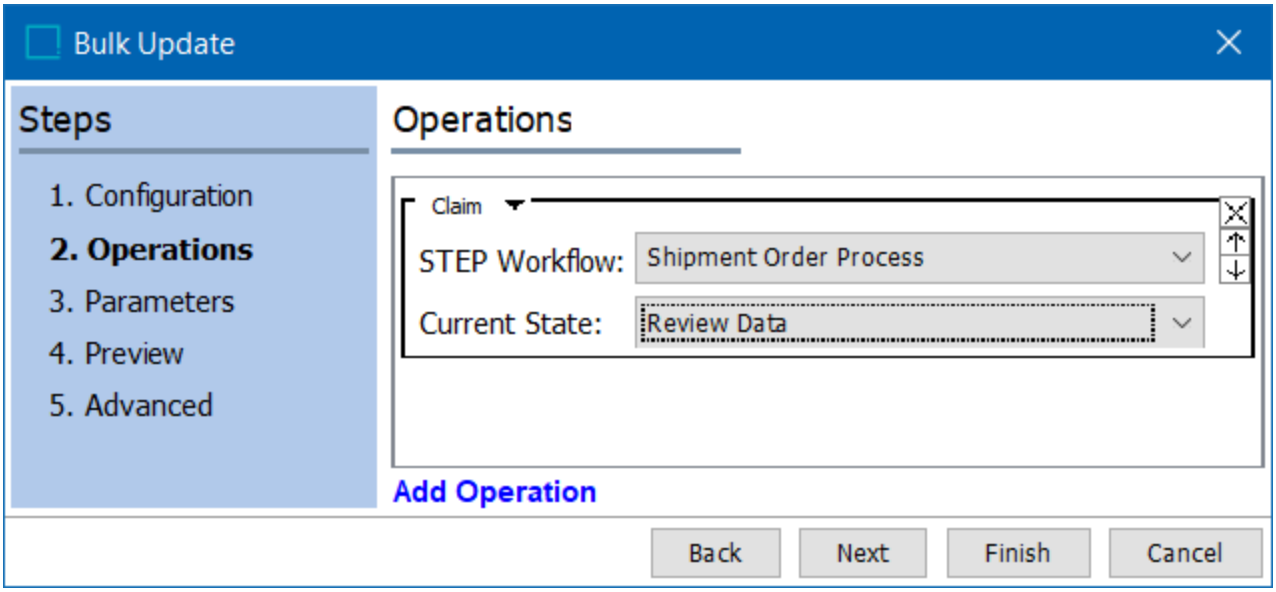


This will create a bulk update for all of the objects in that particular state. Additionally, it is also possible to save the search results as a collection, and schedule a bulk update by selecting the saved configurations. Note that this also saves all tasks that show up in the sate in the collection.

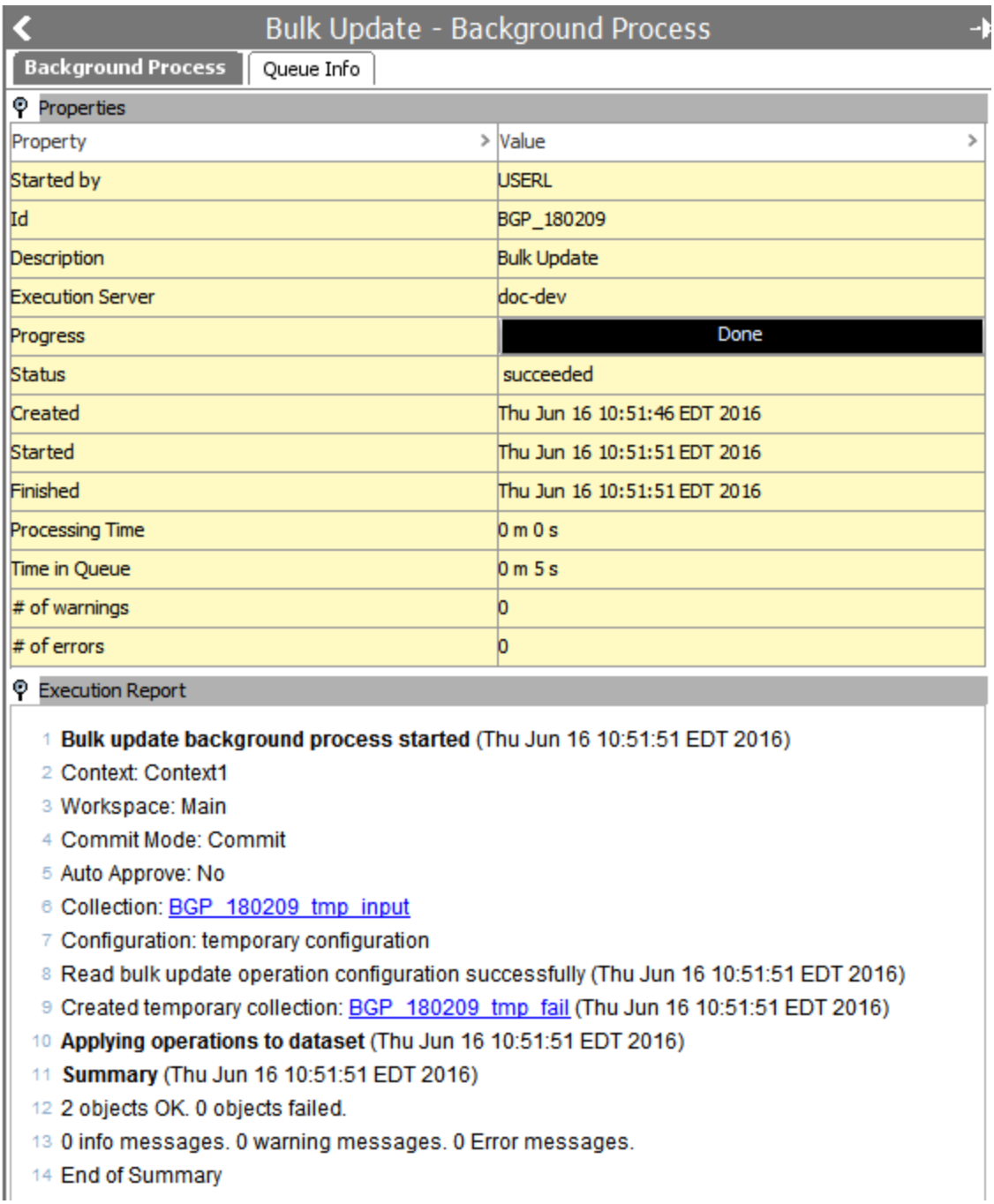


Note: If a user wishes to only claim certain tasks that show up in the search, they should press 'control' and select the desired tasks. Then instead of pressing the bulk update button at the bottom of the search, a user should go up to File > Bulk Update > Run Bulk Update.

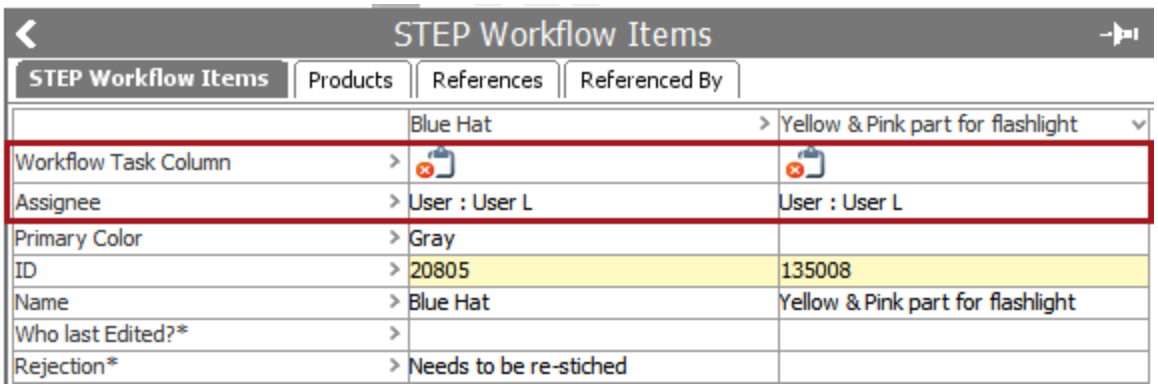
3. Select the workflow and the state from the dropdown.



4. Navigate through the succeeding windows, and click on **Finish**.
5. A background process, BGP, will run and when clicked upon, will display the report of the bulk update process.



6. If the BGP was successful, navigate to the STEP Workflows tab again and check to see that the items are now claimed and assigned to the appropriate user, in this case User L.



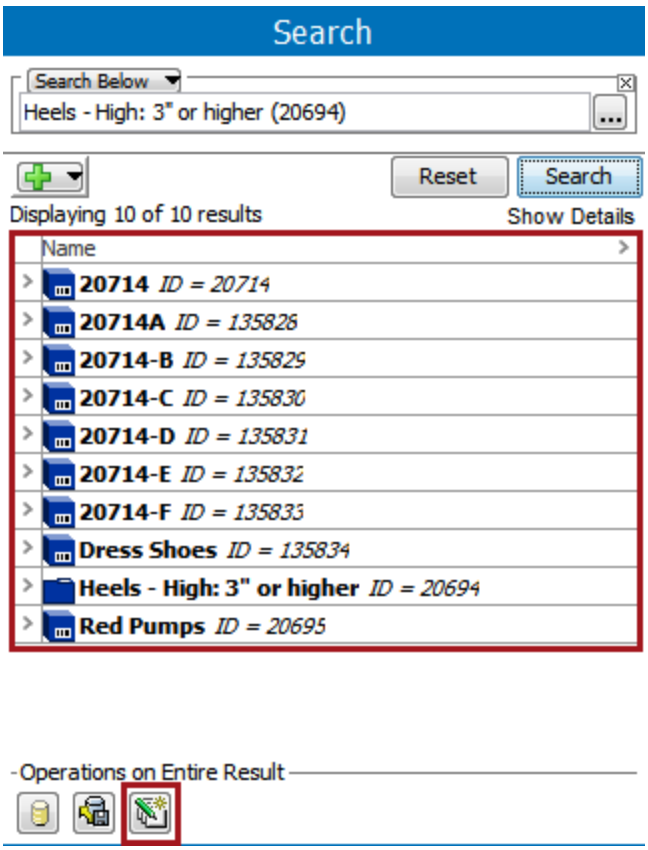
Initiate Items in STEP Workflows

Whenever new objects / products / items / or tasks are created or on-boarded, it has to be processed via the workflow. Instead of initiating each item individually, a user can save time by initiating them all at once in a bulk update.

How to Initiate Tasks in STEP Workflow Using Bulk Updates

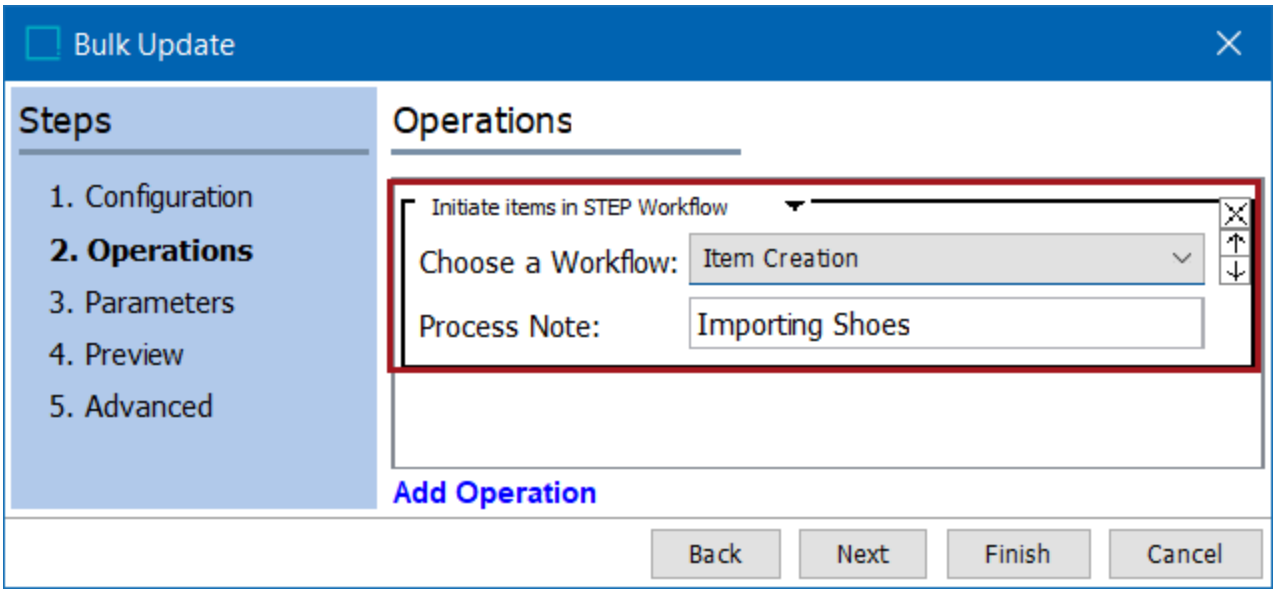
The following steps are required configurations to run a bulk update. It is important to keep in mind that a workflow needs to be valid for the objects / products / items / tasks being initiated into it.

1. First search for the products that are to be initiated into the desired workflow and select the button for a bulk update.

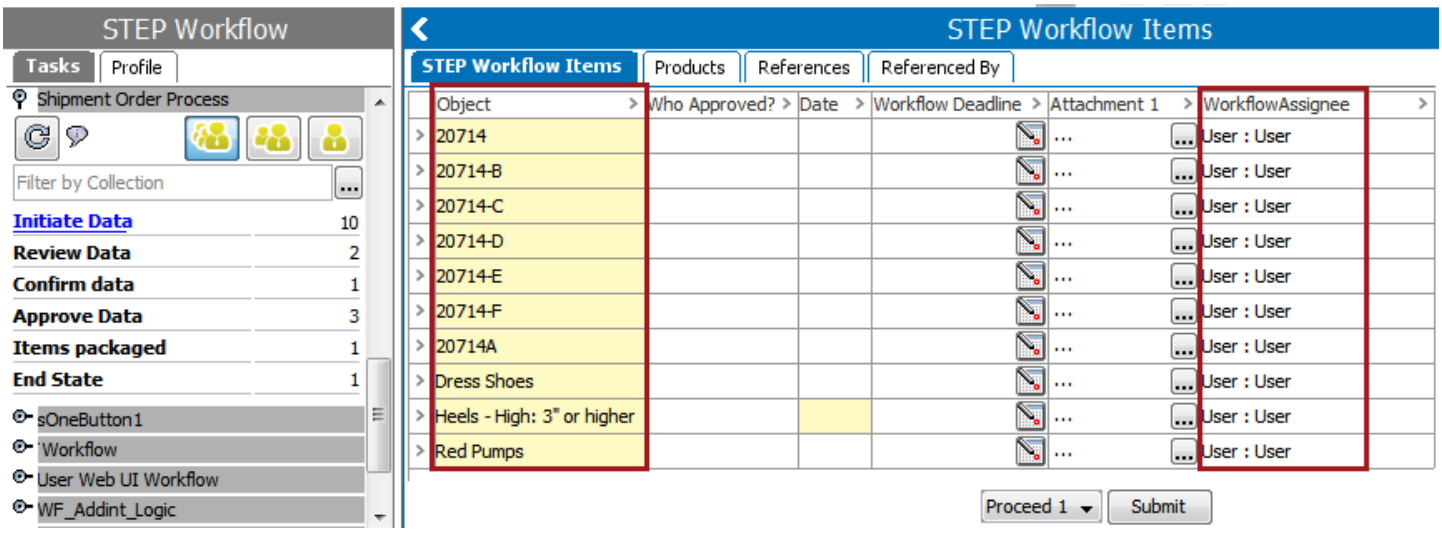


Alternatively, you could click on the objects in tree and go up to File > Bulk Update > Run Bulk Update.

2. In the Bulk Update wizard that appears, navigate to the Operations window and select Workflow > **Initiate items in STEP Workflow**. Select the correct workflow from the dropdown, and add a process note if desired.



3. Navigate through the succeeding windows, and click on **Finish**.
4. A background process, BGP, will run and when clicked upon, will display the report of the bulk update process.
5. If the BGP was successful, navigate to the STEP Workflows tab again and view the items that were initiated in the workflow. The assignee of all of these items will be the user who is logged in and performed the bulk update.



Removing Items from STEP Workflow

Should a user want to remove all objects from one state in a workflow, they can do so using bulk updates if they have the ability to search the STEP workflow tab, and they have the 'Perform Bulk Update' privilege.

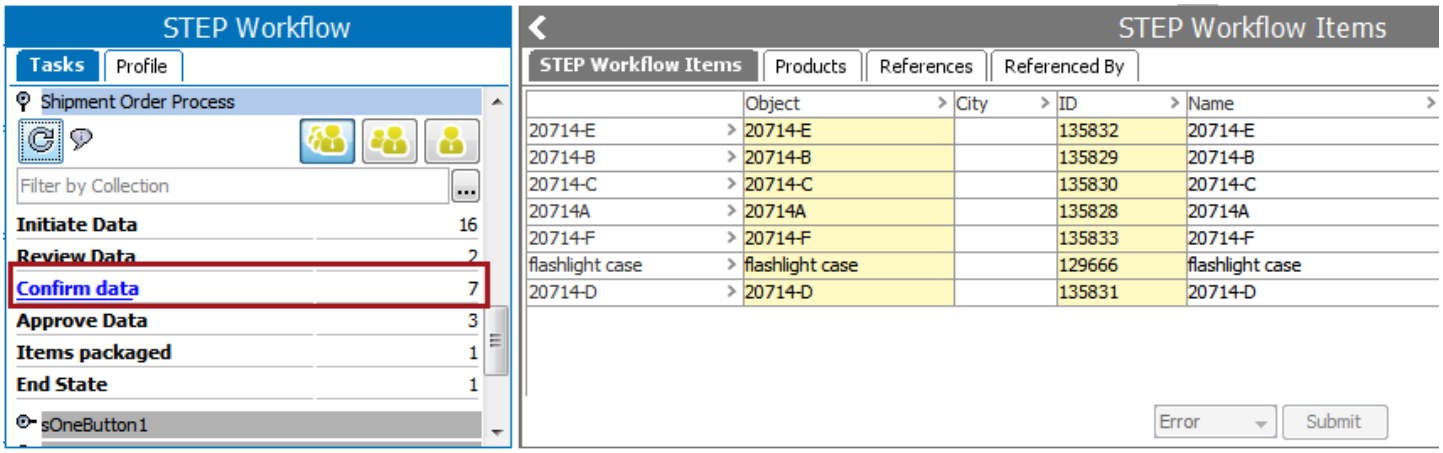
For complete directions on how to use bulk updates to remove items from STEP Workflows, see the **Remove Objects From Workflows** topic in the **Workflows** documentation.

Trigger STEP Workflow Event

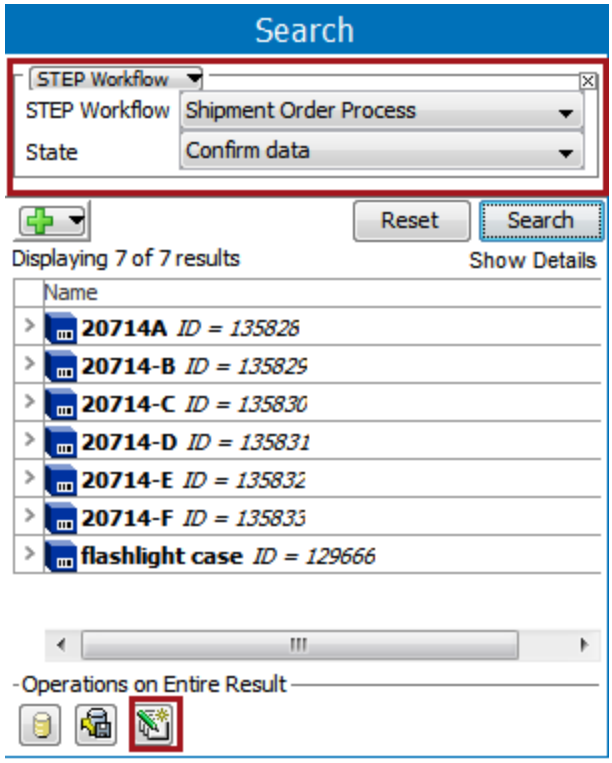
This function is useful when a user completes multiple tasks and needs to trigger multiple tasks from one state to another at one time

How to Trigger STEPS Workflow Events

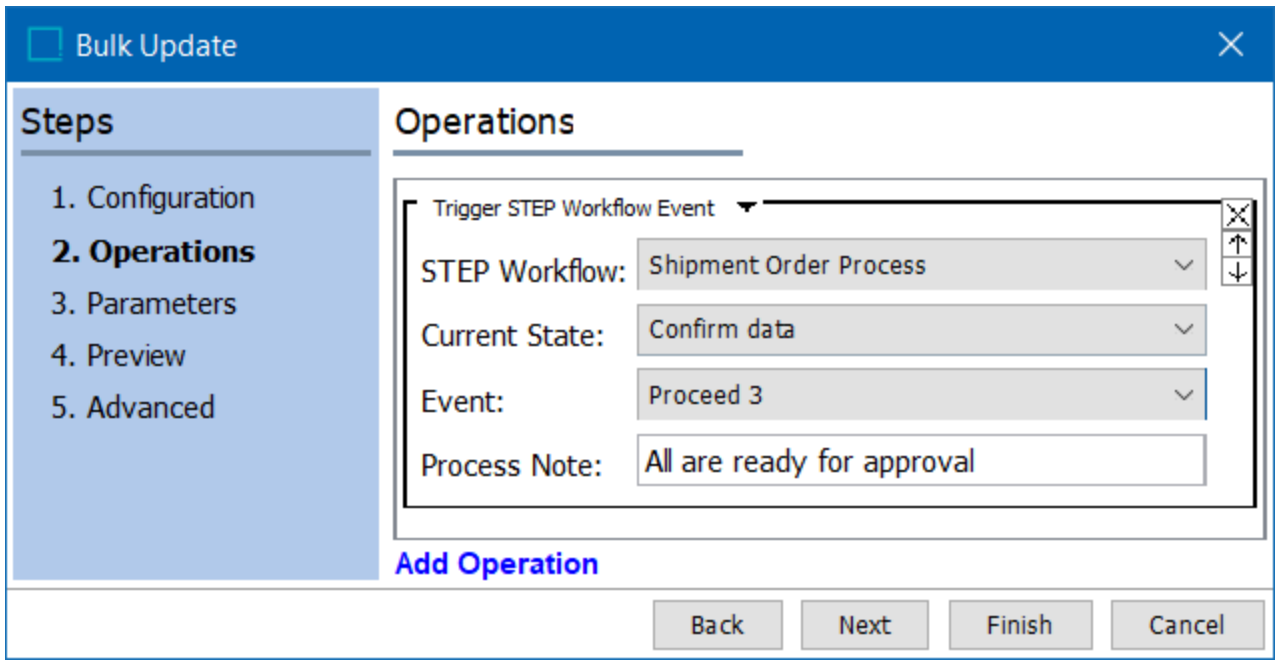
For example, if a workflow has a state called Confirm Data with 7 tasks in it that all need to be moved to the next state, called Approve Data, the user would follow the steps below to perform this action.



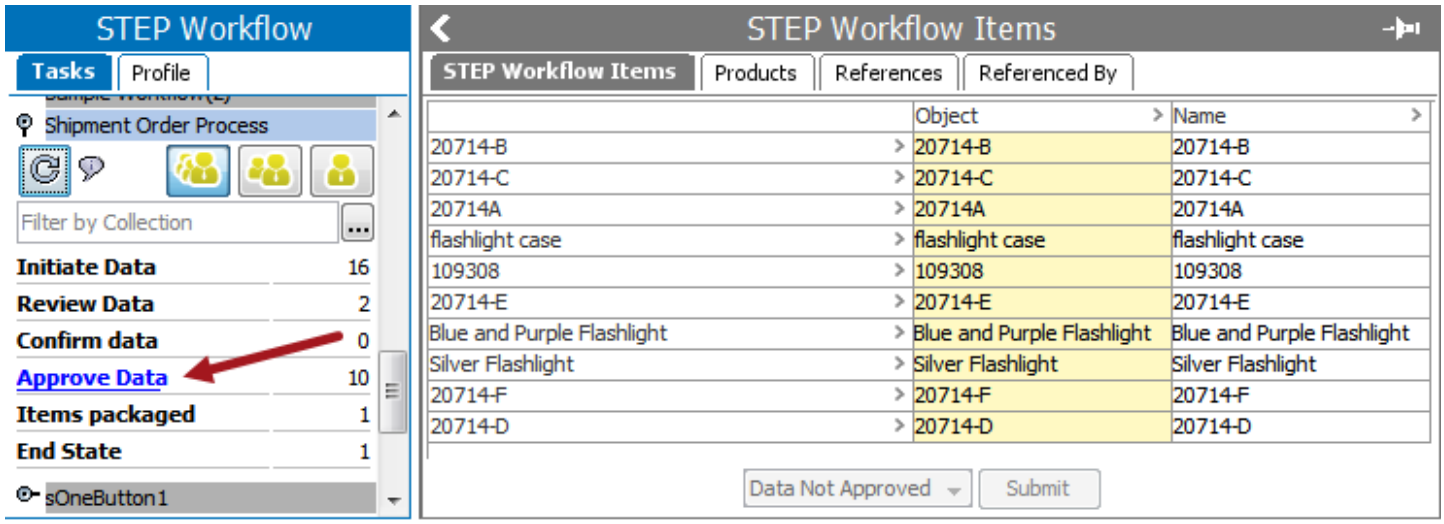
1. First go to the Search tab, and select STEP Workflow from the dropdown. Next select the desired workflow and state to display the products present in that particular state, and click the bulk update button,  .



2. In the Bulk Update wizard that appears, navigate to the Operations window and select Workflow > **Trigger STEP Workflow Event**. Select the correct workflow, state, and event to be performed. Add a process note if needed (process notes are the information for the users in the next state.).



3. Navigate through the succeeding windows, and click on **Finish**.
4. A background process, BGP, will run and when clicked upon, will display the report of the bulk update process.
5. If the BGP was successful, navigate to the STEP Workflows tab again and view the items that were moved from one state to the next in the workflow.



Note: If the Bulk Update is run on objects not in the specified state in the specified workflow, warnings will be generated in the Bulk Update Background Process and the objects left unchanged.

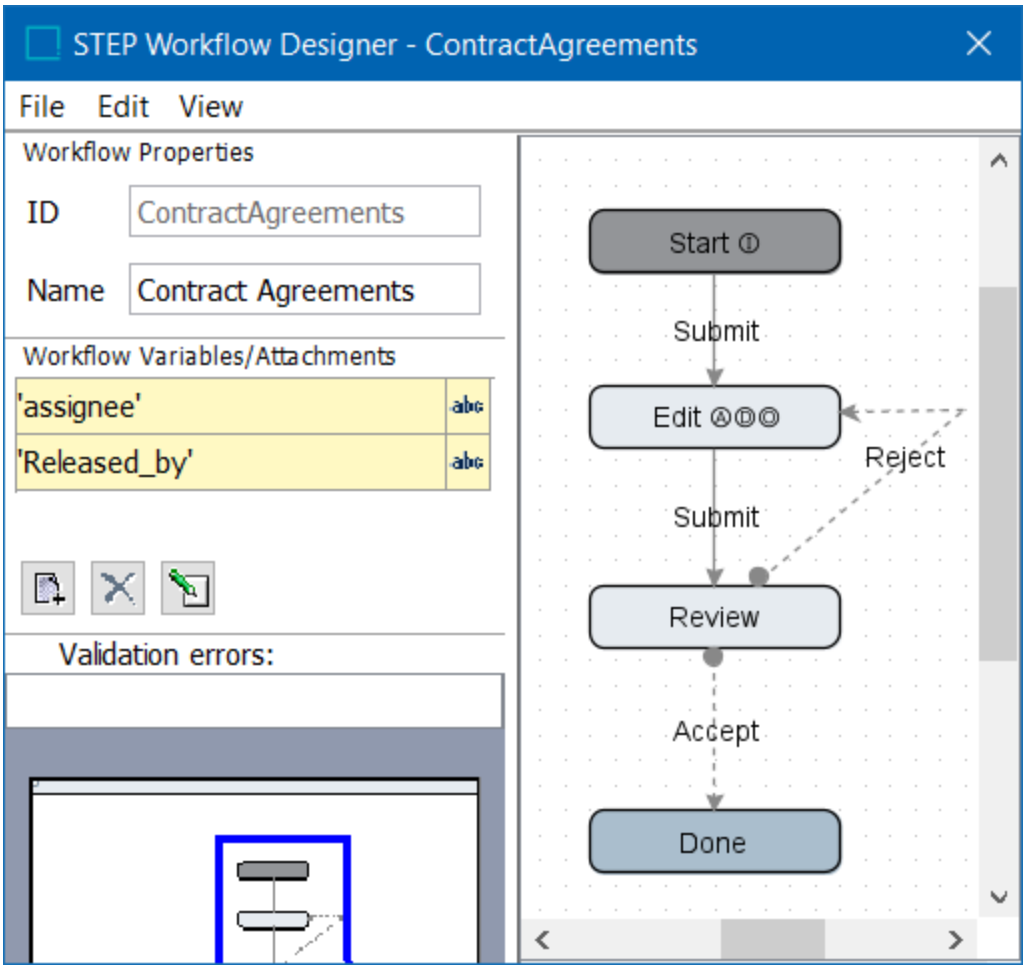
Business Rules and Workflows

When working with workflows, it is very common to use business rules. They can help with a number of functions, such as sending email, validating data, moving a task through a workflow, assigning tasks in a workflow to particular users, among many other actions. It is important to know that business actions can take actions in a workflow, while business conditions are able only to check and/or validate data.

This topic provides a general introduction to some overall functionality specific to business rules *in workflows*. Much more information regarding business rules in general (applicable to rules used both within workflows and external to workflows) is available in the general **Business Rules** section of the documentation.

A Simple Example

Below is an example of how a number of business rules can work together to make a coherent and functioning workflow. It is possible to define business actions to be executed as part of the workflow on states or transitions, and to define conditions to be tested before transitions between states are performed.



For example, a company needs a number of documents signed and agreed upon before they can proceed with their new business partners. In order to make sure that the documents were reviewed and signed to exact specifications, the workflow above was created to help with the process.

For this workflow a document, or task, enters into the Start state where it is picked up by an assigned user, Ⓐ. After all assignments are completed they submit the task to the Edit state. Next, a user claims the task, completes any assignments regarding the task on time, Ⓓ, and then submits it to the Review state. As they submit it to the Review state, a business rule captures the ID of the user who submitted the task, Ⓞ.

Once in the Review state, a new user claims the task. If however, they do not think that it is ready for review, they have the option to send it back to the Edit sate. If they choose to do this, as the task enters into the Edit state again, there are business rules in place to send the task back to the user who originally worked on it in this state, Ⓘ, and sends them an email to notify them it was rejected back to them.

However, if the task enters the Review state and the user who picks it up does think it is ready to be delivered, they can choose to accept it, and submit it to the Done state. Notice that a condition is in place on the transition from the Review State to the Done state. This condition makes it so that the reviewer has to populate an attribute field on the task called 'Approved' to 'Yes' in order for it to be submitted to the Done state successfully.

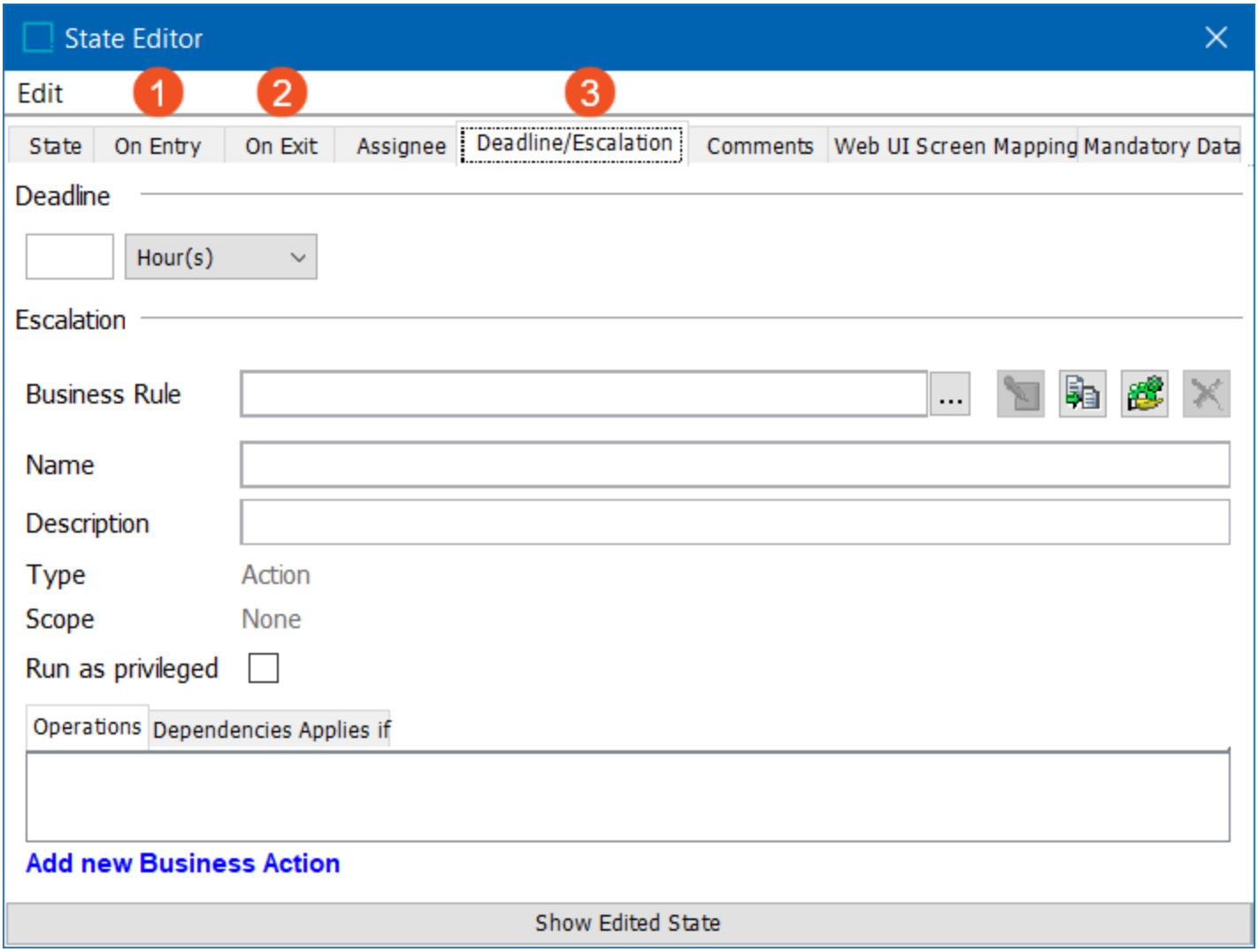
A few aspects to note:

- If a script exception is thrown during the transaction, or the assignee for the Edit state cannot be set, the entire transaction including any changes made is rolled back
- When the On Transition Action is executed the task is not in any state, any attempt to manipulate a task will cause an error to be thrown, and again the transaction will be rolled back

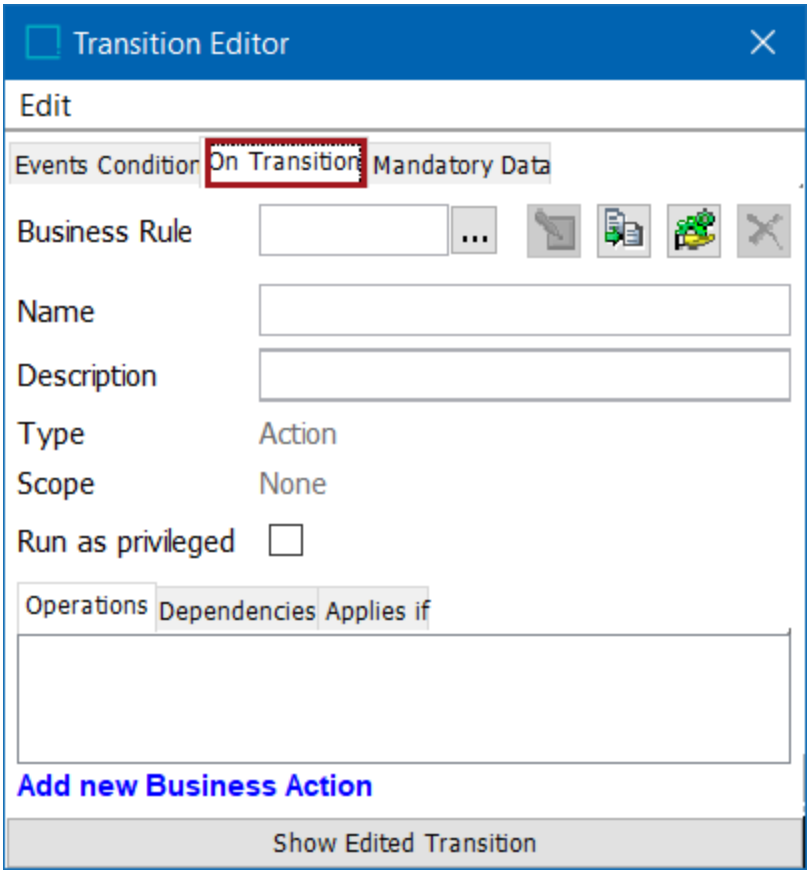
Applying Rules in a Workflow

Business **actions** can executed from workflow **states** at three points:

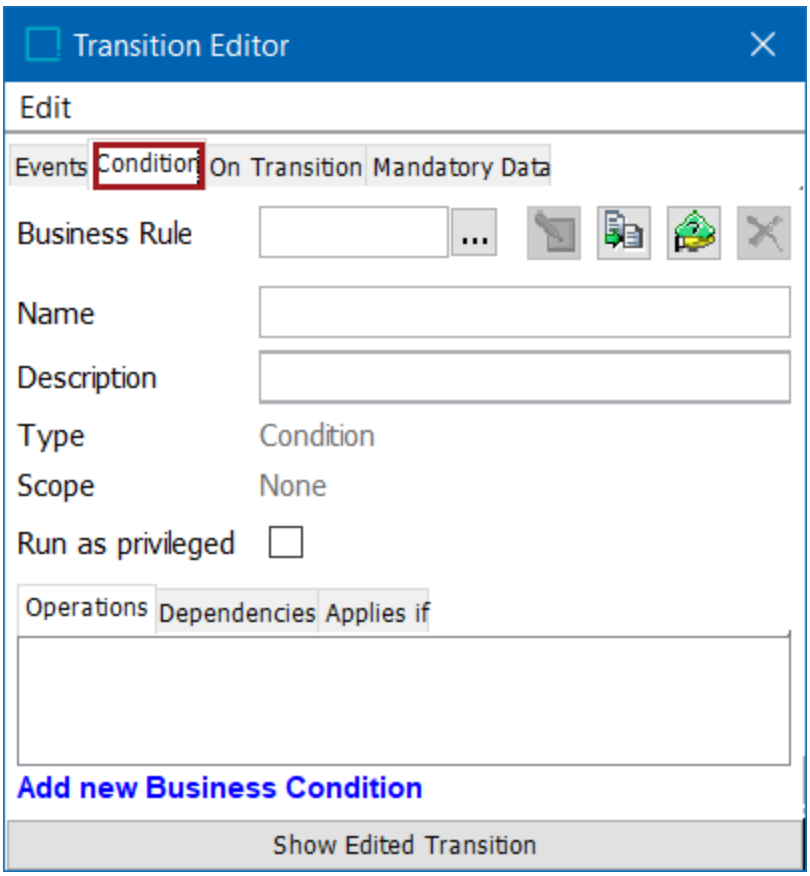
1. Entering a state (On Entry)
2. Exiting a state (On Exit)
3. As part of an escalation when a deadline for a state has been met (see **Deadlines and Escalations** inthe **Advanced Workflow Topics** documentation for more details).



Business **actions** can also be executed on **transitions** between states:

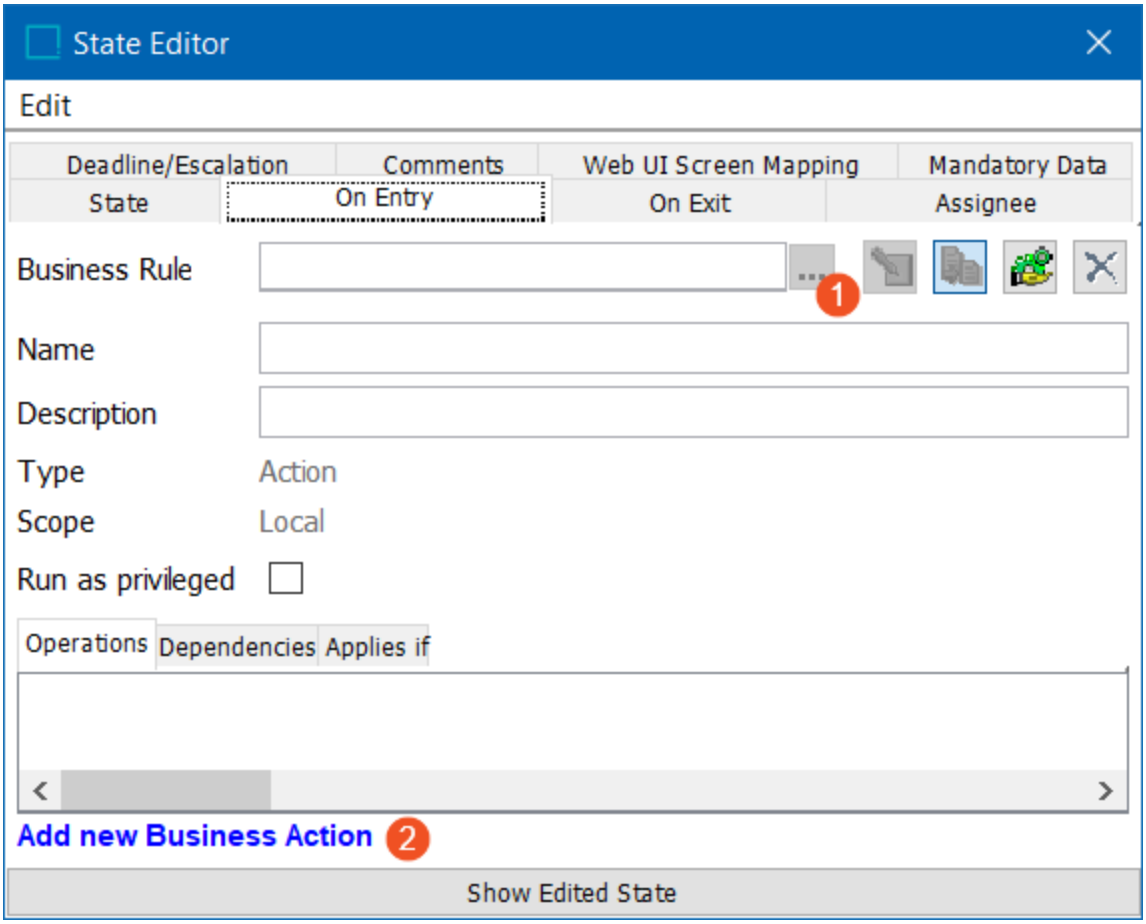


Business **conditions** can be executed only on **transitions**:



Regardless of the place the rule is applied, or whether it is a condition or an action, a few things are common to the editors (and specific to the business rule editors within workflows).

To add a business rule to a workflow, click the ellipsis button (...) and select an existing rule (1), or click the **Add new Business Action** / **Add new Business Condition** link (2).



Defining a new business rule from *within* the STEP Workflow Designer, creates a local business action (2), while selection of an existing rule (1) allows users to choose from global rules.

Important: Local business rules can only be used in the workflow where they were created.

If a rule could be re-used in multiple locations, it is generally better to create it as a global rule.

If a local rule has been created, it can be promoted to a global rule using the hand icon. This creates a global copy of the rule so that it can be used elsewhere.

If a global rule is selected, it can be edited using the edit icon, though care should be take with this as it results in changing the rule in *all* places where it is used.

If a global rule should be changed, it is often better to make a local copy of the rule using the copy button, ensuring that any changes made to the rule will only affect the current workflow.

Local business rules created or edited through the STEP Workflow Designer are saved when the workflow is saved. Global business rules are saved when changes are saved within the business rule editor.

Workflow Business Rules in Operation

Each time a business action or condition is configured on a state or on a transition, the workflow has indicators to signal to the user where these business rules have been applied. The indicators are the following:

Indicator	Description
'A'	A default assignee for a particular state (or workflow) has been assigned.
'D'	A default deadline has been selected.
'I'	Signals the configuration of an 'On Entry' action. It may help to remember this as 'I' for 'In' (e.g., into the state).
'O'	Signals the configuration of an 'On Exit' action. It may help to remember this as 'O' for 'Out' (e.g., out of the state).
dotted transition line	Indicates a condition has been placed on the transition.
rounded end on transition	Indicates an action has been placed on the transition.

Order of Operation

It is important to note the order of operation of the various rule types (condition or action) and where they are applied (entry, exit, transition, or deadline). Consider an object going from a current state to a next state. Business rules are evaluated in this order:

1. Condition on transition exiting current state
2. On exit action from current state
3. On transition action between current and next state
4. Default assignee on entry to next state (if assignee is set in workflow designer rather than using business rules)
5. Deadline set in next state
6. On entry action in next state

More Information

This topic has provided a basic introduction to how business rules can be used in conjunction with workflows. Topics covering how to carry out all of the functions described in the above example can be found throughout the **Advanced Workflow Topics** section of the documentation, in addition to other use cases and topics, including:

- Assigning Tasks Using Business Rules
- Auto-Routing Tasks in a workflow
- Configuring Status Flags in workflows
- Deadlines and Escalations in workflows
- e-Signature used to re-authenticate in a workflow
- Java API for workflows
- Remove Objects from workflows
- Running Background Processes from workflows
- Send Email from a workflow
- Validate Data in a workflow

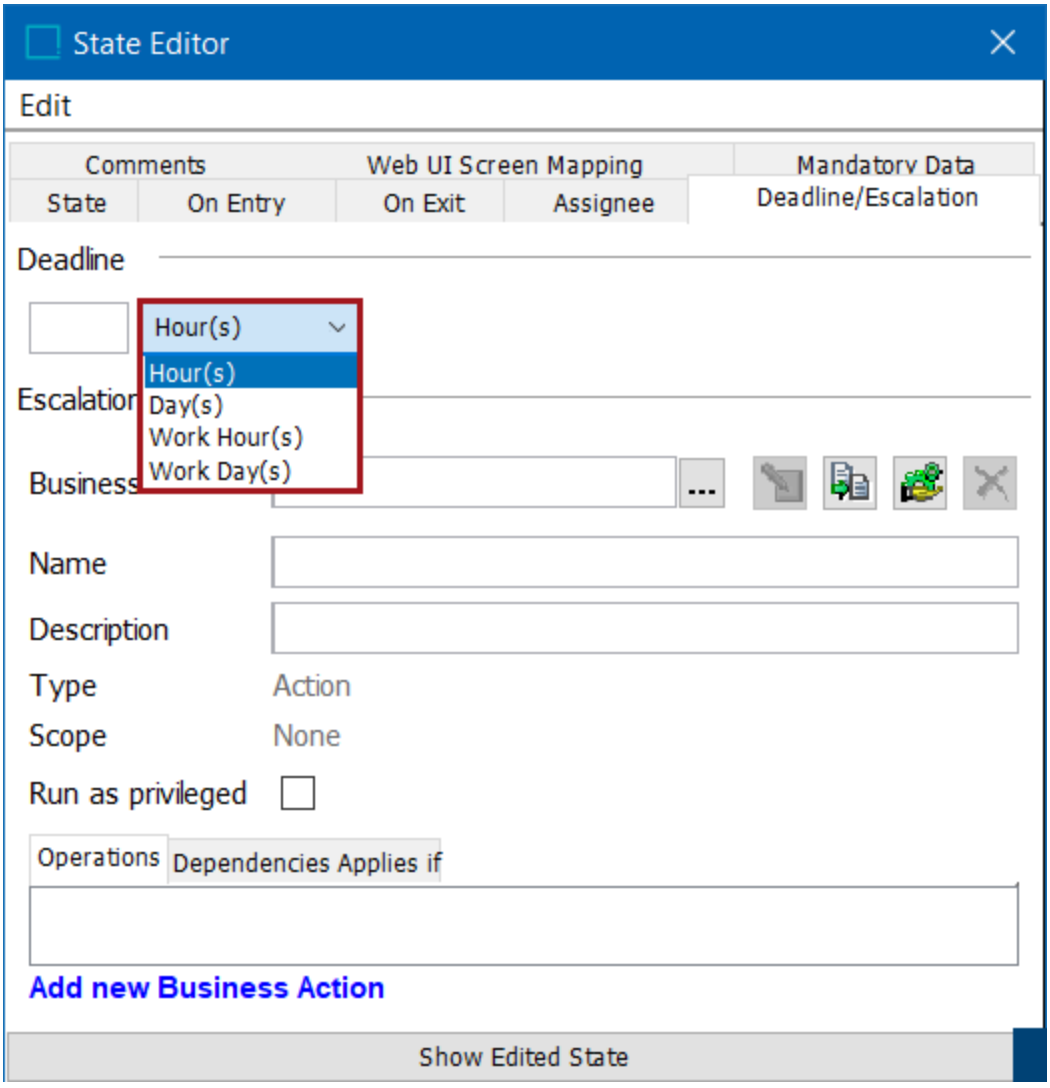
As business rules provide robust functionality to meet the individual needs of each customer, it will also be useful to see the general **Business Rules** section of the documentation to understand more advanced features, as well as the Scripting API section of the STEP API documentation, accessed by clicking the **STEP API Documentation** button on the STEP Start Page.

Deadlines and Escalations in Workflows

In the Workflow Designer a user can define a deadline for a state via the State Editor Deadline/Escalation tab. This is useful for a number of reasons, one of which might be to notify a supervisor, or task manager, if a deadline is not met for a particular task in a particular state. This may be important to know because if the task is late, it could impact many steps that come after it.

Understanding Deadlines in the Workflow State Editor

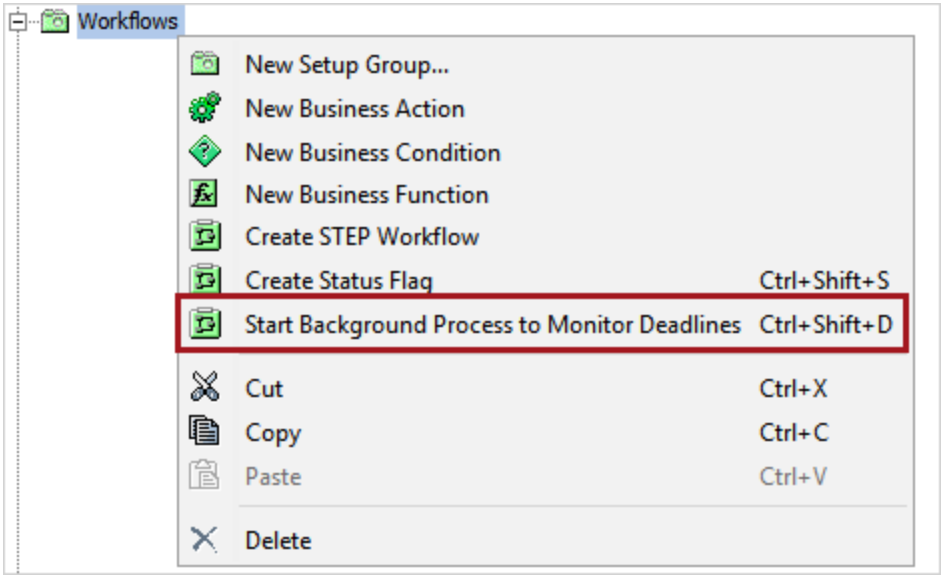
The deadline is set relative to the time an object enters the state. For example, if the deadline is set to '5 Hour(s)', an object entering the state at 6 a.m. will get the deadline set to 11 a.m. The available time periods are: Hour(s), Day(s), Work Hour(s), and Work Day(s). Work hours are hard-coded to be from 8 a.m. to 4 p.m., with work days being Monday to Friday (bank holidays and the like are not taken into consideration).



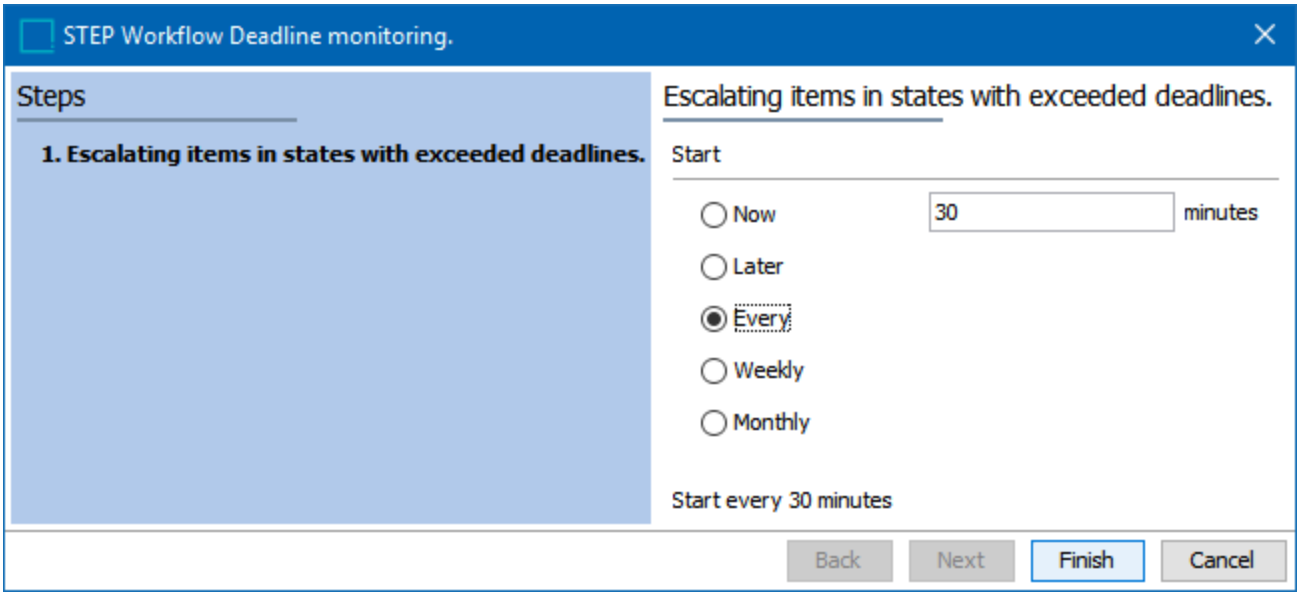
Setting Up Deadlines for Workflows

1. In System Setup, navigate to the desired workflow, right-click on it, and select Edit STEP Workflow.
2. Right-click on the desired state and select Edit State.
3. In the new dialog, select the Deadline/Escalation tab.

On the same tab it is possible to define or select a business action that should be executed if the deadline is met and the background process for monitoring deadlines is run. To run this background process, right-click the Workflows folder in System Setup, then select 'Start Background Process to Monitor Deadlines.'



Once selected, the dialog to set the frequency to monitor deadlines displays.



This background process (BGP), which can be scheduled to run at specific intervals, can be started from any Setup Group in System Setup that can hold workflow objects. Since this initiation of the BGP happens at the folder level, this setting applies to all workflows contained in the folder.

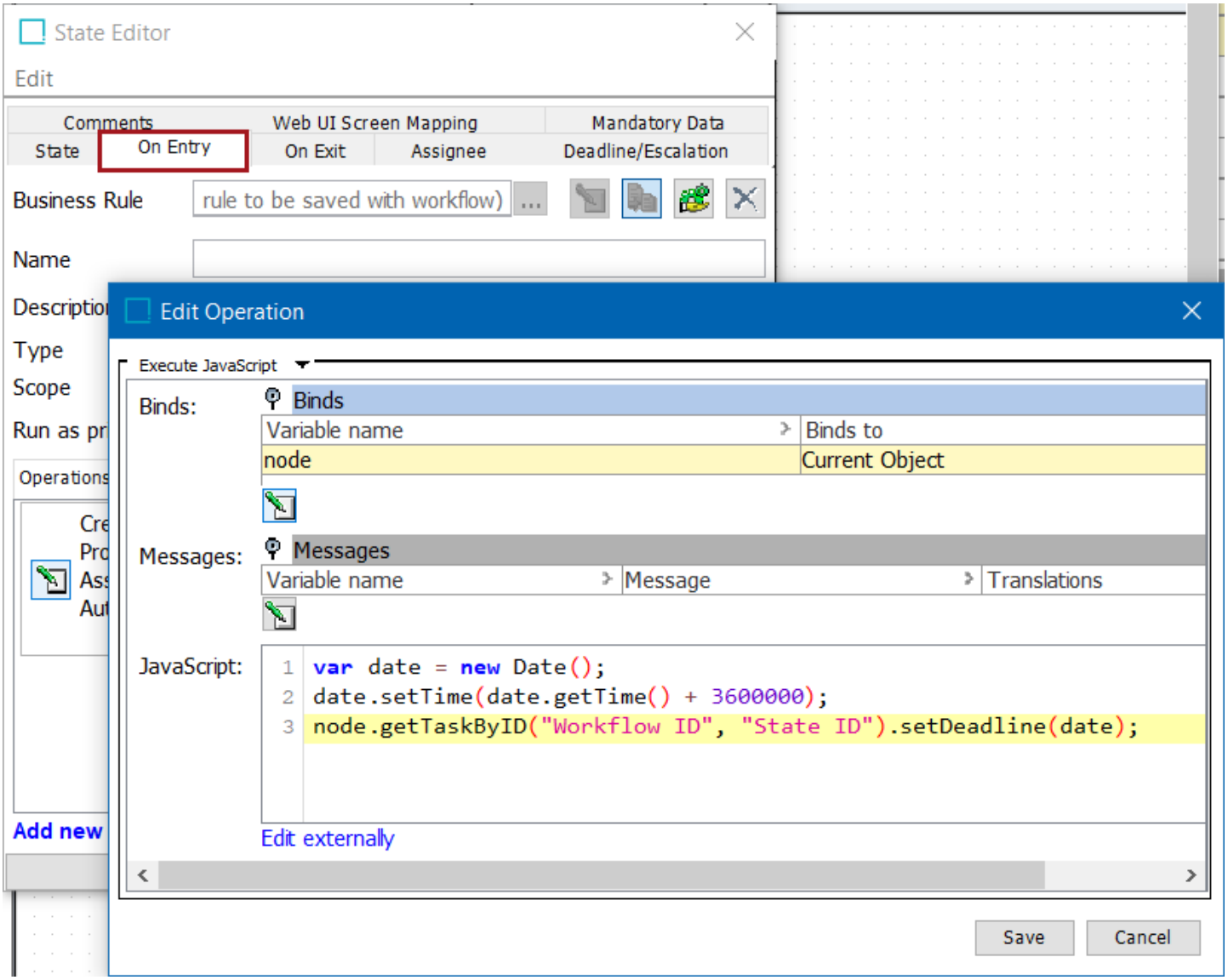
Note: Once the deadline has elapsed, the escalation rules will run every time the BGP set in the previous image looks for passed deadlines. For example, with a four-hour deadline and an 'Escalating Items' BGP set to start every 30 minutes, the escalation rules run with four hours have elapsed, and then again every 30 minutes until the object in this escalated status is moved to a new state.

Business Actions for the Workflow On Entry Tab

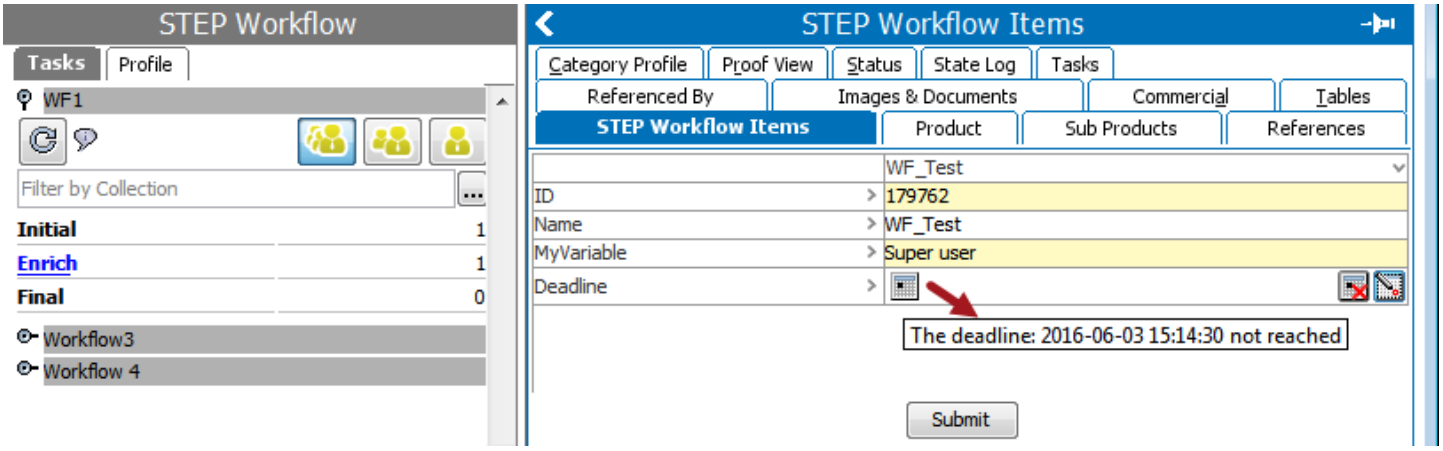
As described in the **Views and Mappings for Workflows** section of the **Workflows** documentation. Deadlines can be viewed and manipulated in views via the 'WorkflowDeadline' Component. In the public API, methods for getting and setting deadlines are available via the Task interface. Below is an example of how a deadline can be set to an hour from now in a business action, in an On Entry tab.

```
var date = new Date();

date.setTime(date.getTime() + 3600000);

node.getTaskByID("Workflow ID", "State ID").setDeadline(date);
```

As the deadline is set via the Task interface, the above will only work from the state with ID 'State ID'. An example can be seen in the image below:



When viewed in STEP Workflow, if a deadline is not yet met it looks like the image below:



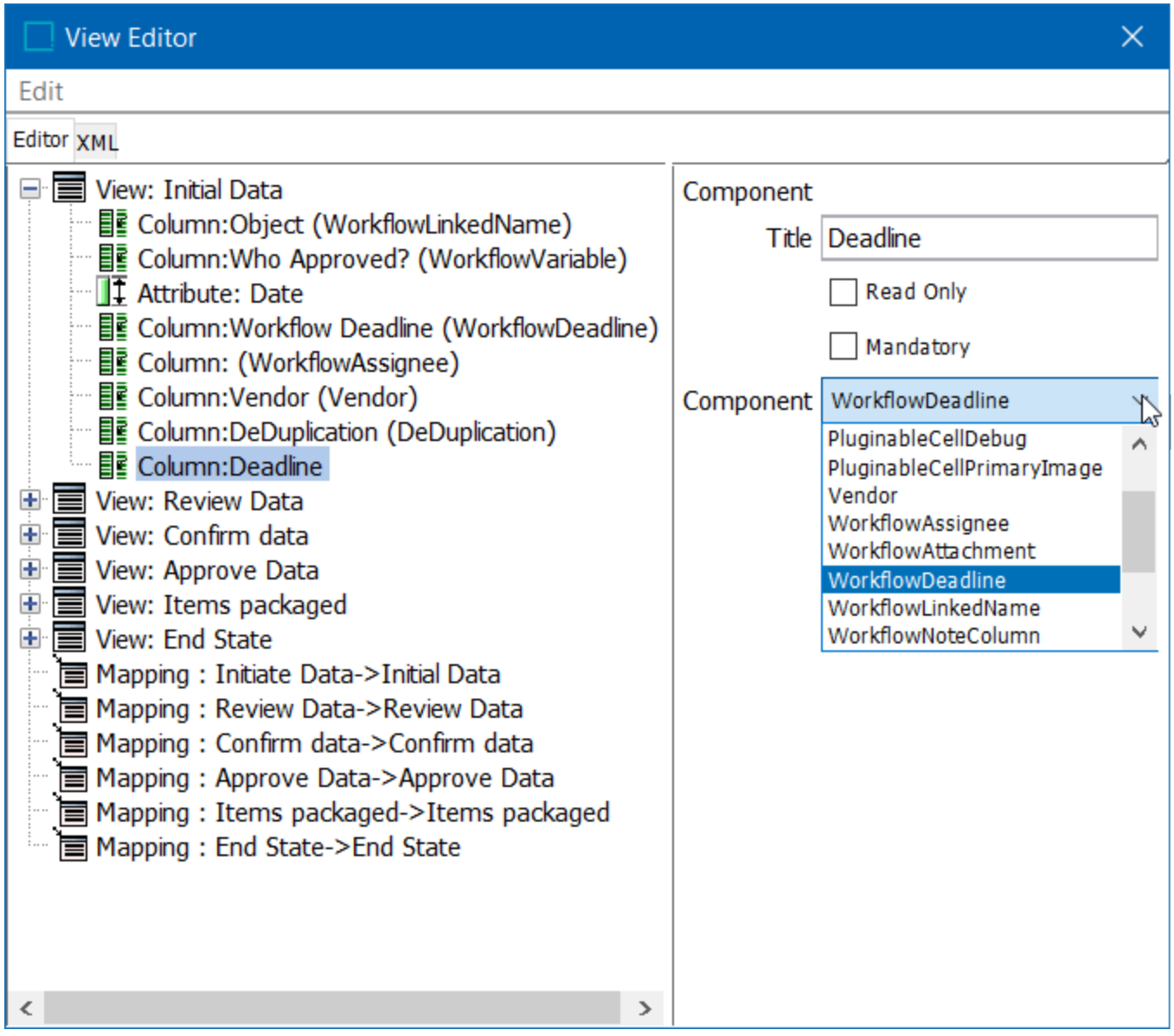
If a user wishes to remove a deadline from a business rule, this can be done by setting the deadline to null, as shown below.

```
node.getTaskByID("Workflow ID", "State ID").setDeadline(null);
```

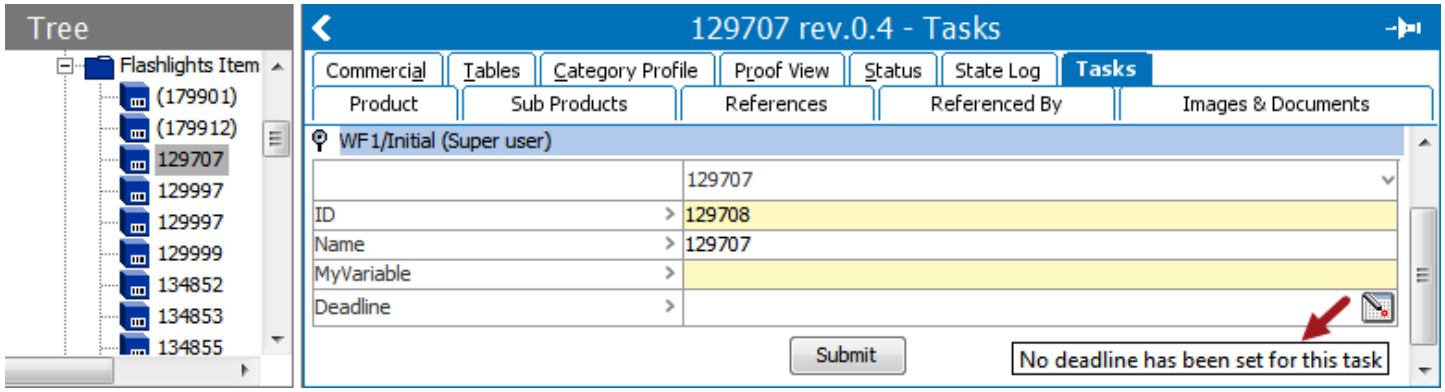
Editing Deadlines through Workflow Views and Mappings

Deadlines can also be set by editing the workbench views and mapping for a given workflow. To do so:

1. Open the desired workflows Designer and go to Edit > **Edit Workflow Views and Mappings**.
2. In the view that needs a deadline added to it, right-click on the view and select **Add Column**.
3. In the Component dropdown, select **WorkflowDeadline**.



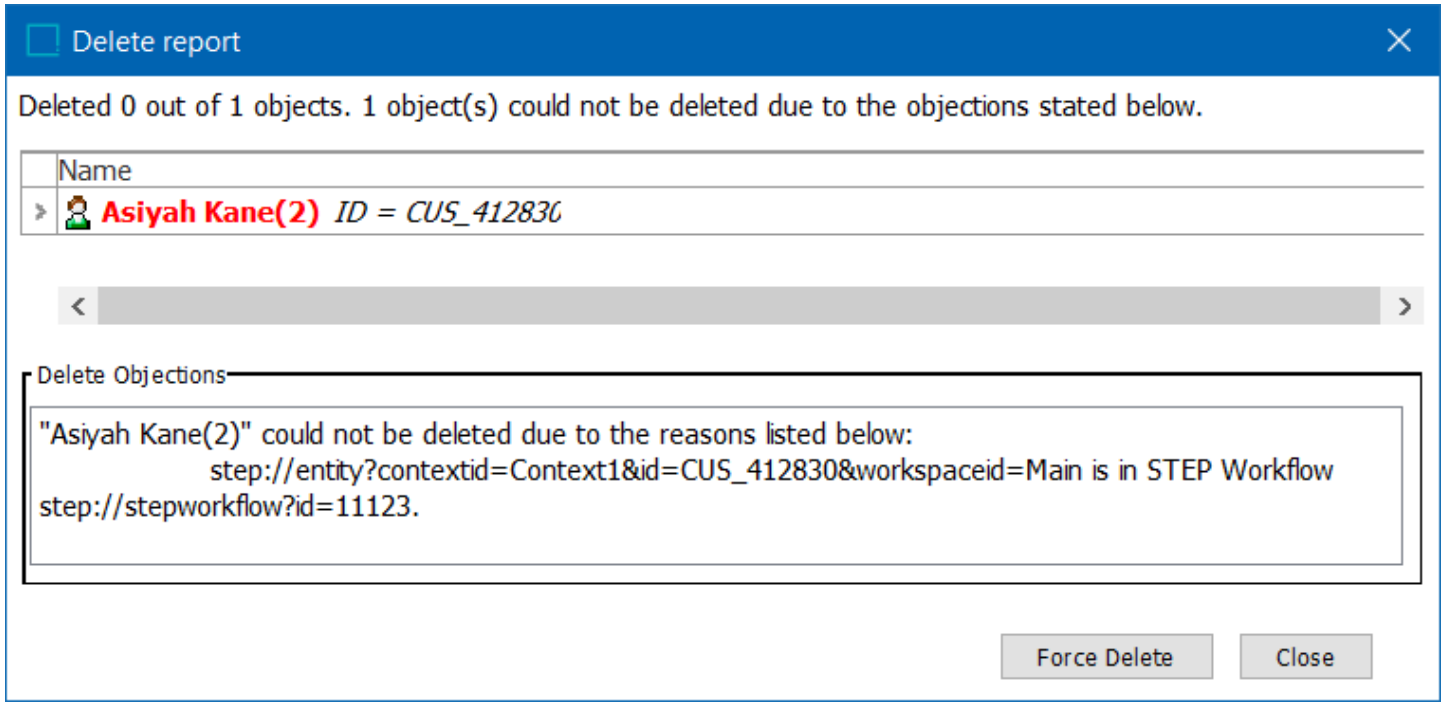
Once selected, the deadline can be set manually in the object itself via the object's Task tab, or in the STEP Workflows navigator tab.



Note: State Editor Deadline/Escalation tab allows a user to set the deadline for all the products in a particular state within the workflow, but the deadline set using workbench views and mappings will apply for a particular product in a particular state in a workflow.

Deleting Objects in Workflows

It is sometimes necessary to delete a product that is in a workflow. This could be for many reasons, including that the product is no longer needed or is discontinued. However, if the product is currently in a workflow and a user is trying to delete an object in a workflow, the user will be prohibited from doing so unless they 'force delete' the object.



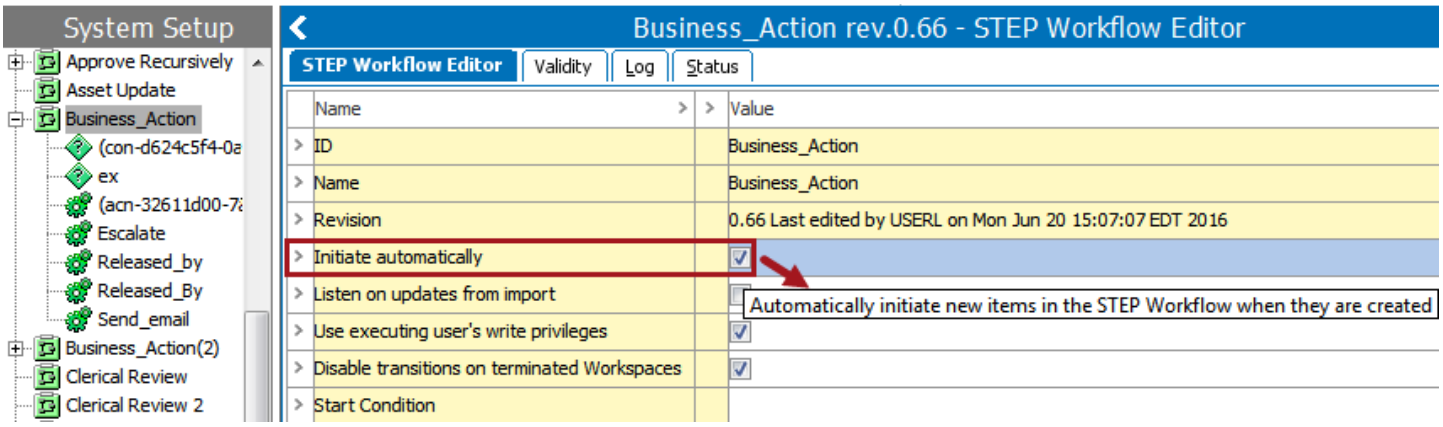
The above message will include the list of workflows in which the product or entity exists. If the user chooses to force a deletion, it will not only delete the product / entity from the hierarchy but will also take it out from all workflows it is currently in.

Note: For objects that are not deleted directly, but put into the Recycle Bin, they will not be inserted into the states they were in prior to the deletion if revived.

A force deletion of a product that is currently in a workflow should only be done when a user is sure that the product / entity is no longer required.

Should a user move a product / entity to the Recycle Bin, this will also take the product / entity out of any workflows. However, if the product / entity needs to be revived from the recycle bin it should be known that once revived, it will not automatically go back into any workflows and their states that it was in initially. If it is desired that

the product / entity that was revived from the Recycle Bin go back into any workflows it once was in, then the option 'Initiate automatically' in System Setup on the STEP Workflow Editor tab needs to be checked for all the workflows that need that particular product / entity again.



If auto-initiation is turned on for the workflow, a new instance will be created. This means that if a product / entity has an object type which is valid for a particular workflow, then the product / entity is automatically initiated to that workflow at the beginning.

Edit a Workflow in Use

Once a workflow has been created and is in use, it may be necessary to change it. If the workflow is not yet in use (e.g., there are no active tasks for the workflow, or the workflow is in a test state only and not yet released for production), editing a workflow is does not require any special considerations.

General information on creating and editing workflows can be found in the **Getting Started with STEP Workflows** documentation, specifically in the **Creating a Workflow** and **STEP Workflow Designer** topics.

This topic provides guidance specifically on editing workflows that are already in production and/or contain active tasks.

Important considerations before editing a workflow in use

- When there are objects in a workflow, certain changes to the workflow are prohibited as they could cause disruption to the tasks within them. Thus it is not possible to rename, delete, or change the state characteristics of any states that have tasks within them. In addition, states with active tasks cannot be moved inside or taken out of a cluster or parallel, and parallels may not have new clusters added.
- It is best to receive advice and guidance from a STEP expert before making changes to a live workflow.
- It is always recommended to duplicate the existing workflow, do the necessary modifications in the duplication, test it, and then apply the same changes to the workflow in use.
- If the required changes are significant, it may be best to process all existing tasks in the workflow, and make sure that no new tasks are added, before the changes are made to the workflow.
- Already existing business rules should be taken into consideration when editing a workflow. It is possible for a rule to rely on a state that is deleted, event that is changed, etc that could prevent the rule from working following updates to the workflow. It is important to evaluate all existing business rules following changes to a workflow, and to test those that may have been impacted.

Exporting Workflow Definitions

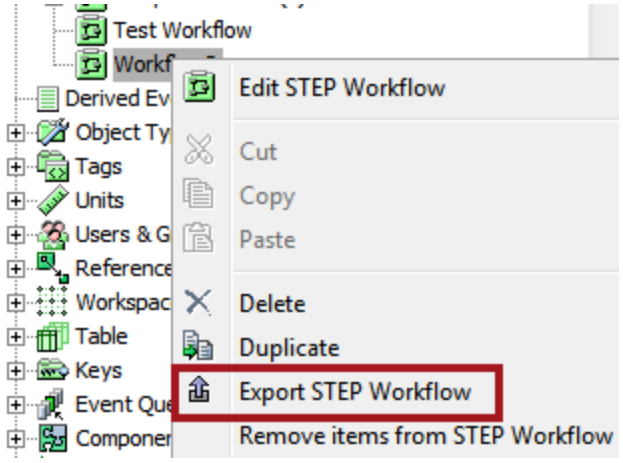
In order to move workflow configurations between systems and/or to compare workflow definitions using external source control systems, workflow definitions can be exported using STEPXML and Advanced STEPXML. Depending on the intended use of the output file, the appropriate method should be selected.

Exporting Workflow Configurations for Loading to Another System

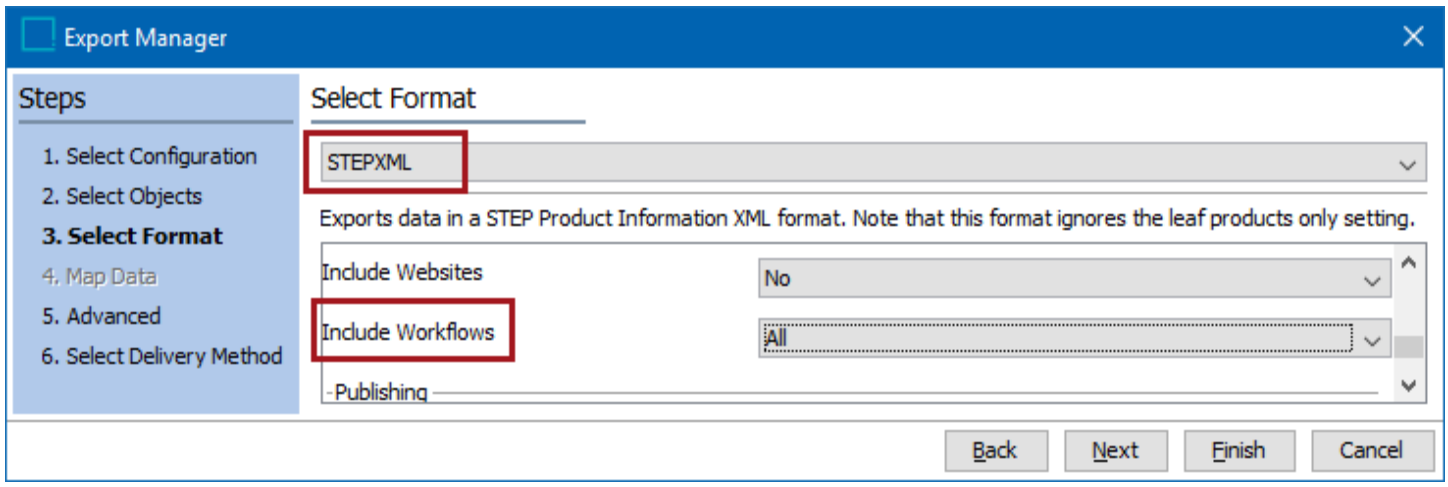
Workflow definitions (including global and local business rules used by the workflow) can be exported in several ways. The simplest of these is described. Exports carried out in this manner provide an encoded workflow definition that, while not necessarily human-readable, provides all necessary information to replicate the workflow on another STEP system.

A workflow definition (including used global and local business rules) can be exported in XML.

1. Right-click on the workflow object in System Setup and selecting **Export STEP Workflow** from the menu.



2. In the pre-configured Export Manager wizard, proceed to the Select Format step and select **STEPXML** from the dropdown.



3. Set the Configuration > **Include Workflows** parameter to 'All' so that the selected workflow is exported.
4. Set the Configuration > **Include Business Rules (Global) and Libraries** parameter to 'Referenced' so that all global business rules used in the workflow will be included in the export.

   Note: Local business rules are exported nested in the STEP Workflow element and will always be included.
5. Navigate through the wizard, making any other required settings, and click **Finish**.

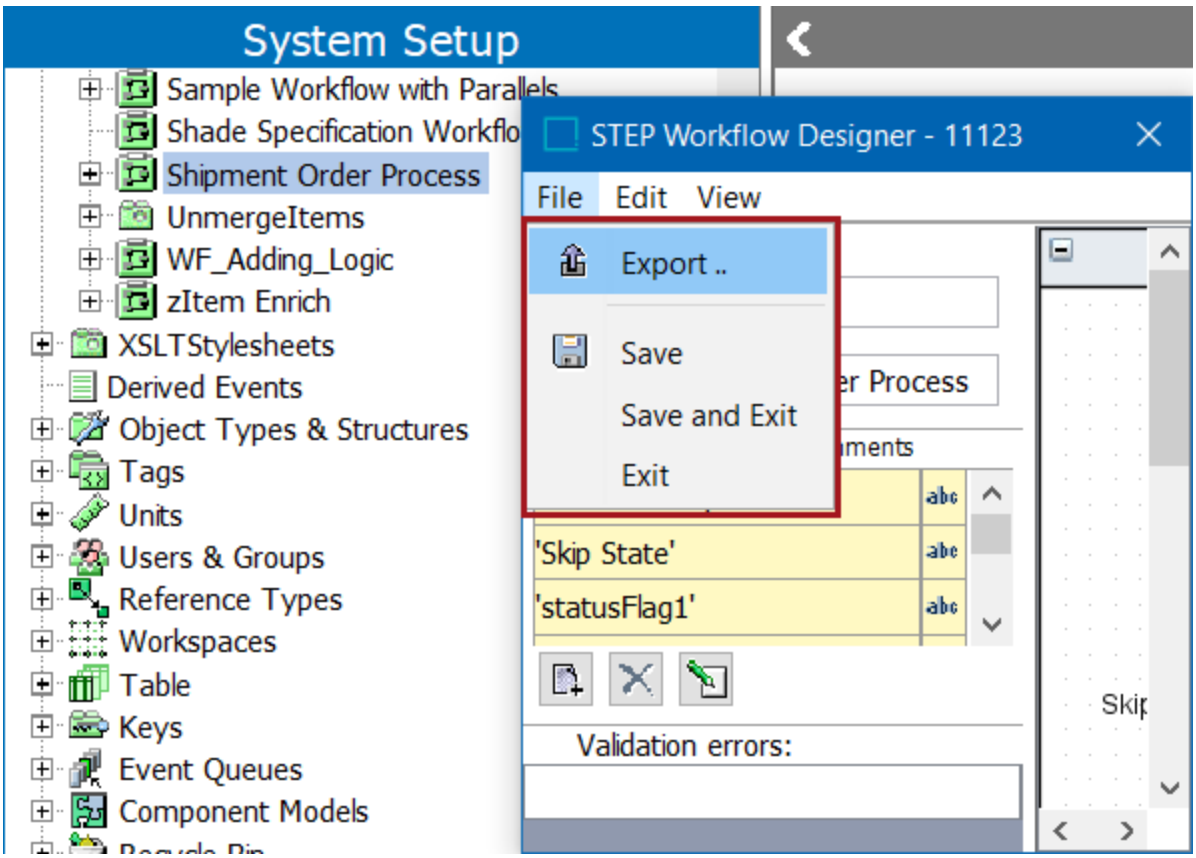
The exported STEPXML file can be imported on another STEP system via the standard Import Manager.

The same export can also be accomplished using Advanced STEPXML. Following the steps outlined above, but selecting the 'Advanced STEPXML' format on the **Select Format** step of the Export Manager will yield a pre-populated export template that will produce the same results.

Note: The export option on the workflow designer File menu lets you export the SCXML on which the workflow is based. This option is for debugging only and should not be used to transfer workflow configurations between systems.

Steps to export the workflow in the SCXML format

1. Go to System Setup and locate the workflow
2. Right-click on the workflow and then click on Edit STEP Workflow option
3. In the new window, click on File -> Export, and then select the path in which the file is to be saved



Exporting Workflow Definitions as Comments for External Comparison

Workflow definitions can be exported as comments using Advanced STEPXML. These exports can be submitted to external source control systems for comparison purposes. Exports generated in this manner are not encoded (as described above), allowing for readability by other systems. However, these exports do not provide the necessary information to transfer a workflow between STEP systems and use in this manner is not supported.

To export a workflow for external comparison, Advanced STEPXML must be used and the DefinitionsAsComments tag must be set to 'true'.

On the Select Format step of the outbound tool, choose the Advanced STEPXML format, then copy and paste the following text into the Template field:

```
<?xml version='1.0'?>
<STEP-ProductInformation DefinitionsAsComments="true">
<STEPWorkflow ExportSize="All"/>
</STEP-ProductInformation>
```

An example of the output for a workflow definition as comments is shown below:

```
<STEPWorkflow...>
<!--Definition:[This is the Workflow definition. Removed for brevity.]-->
```

```
[Remaining configuration]
</StepWorkflow>
```

Note: The content of the comment field is not part of the STEPXML XSD and therefore Stibo Systems reserves the right to change the format of the output content at any time.

For more information, see the **STEP-ProductInformation Tag in STEPXML** section of the **Data Exchange** documentation.

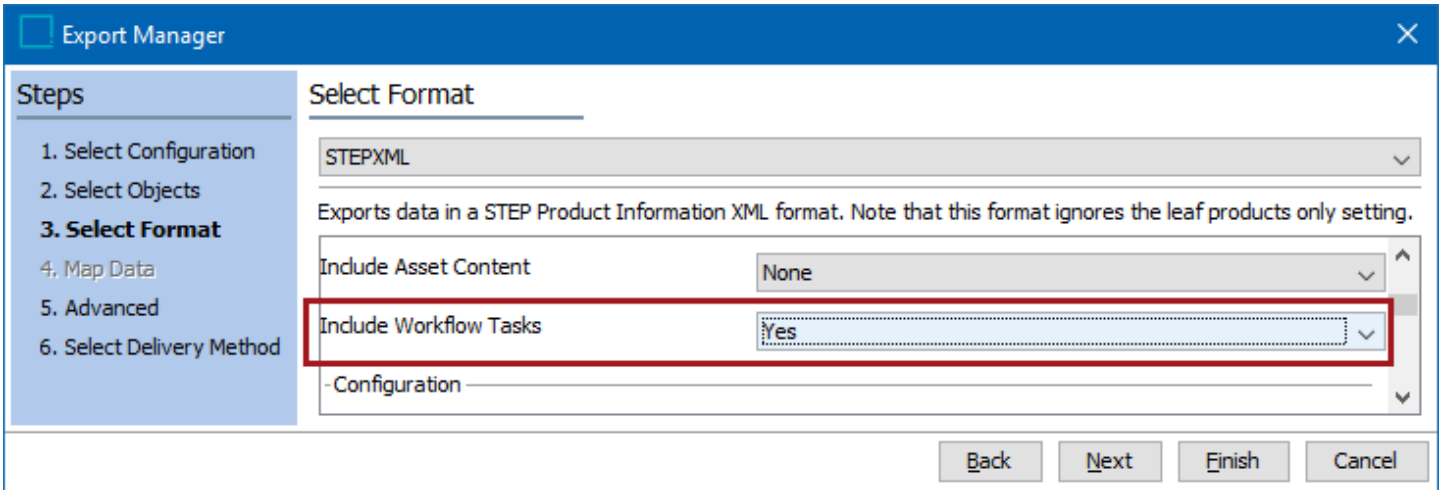
Exporting Workflow Statuses

When exporting Products, Classifications, Assets, and Entities it is possible to have workflow related information included in the exported file.

Workflow status export (STEPXML or Excel) is a kind of meta data information to communicate with the other systems. For example, in the case of translation XML, the workflow states will act as a flag to match the status in other systems.

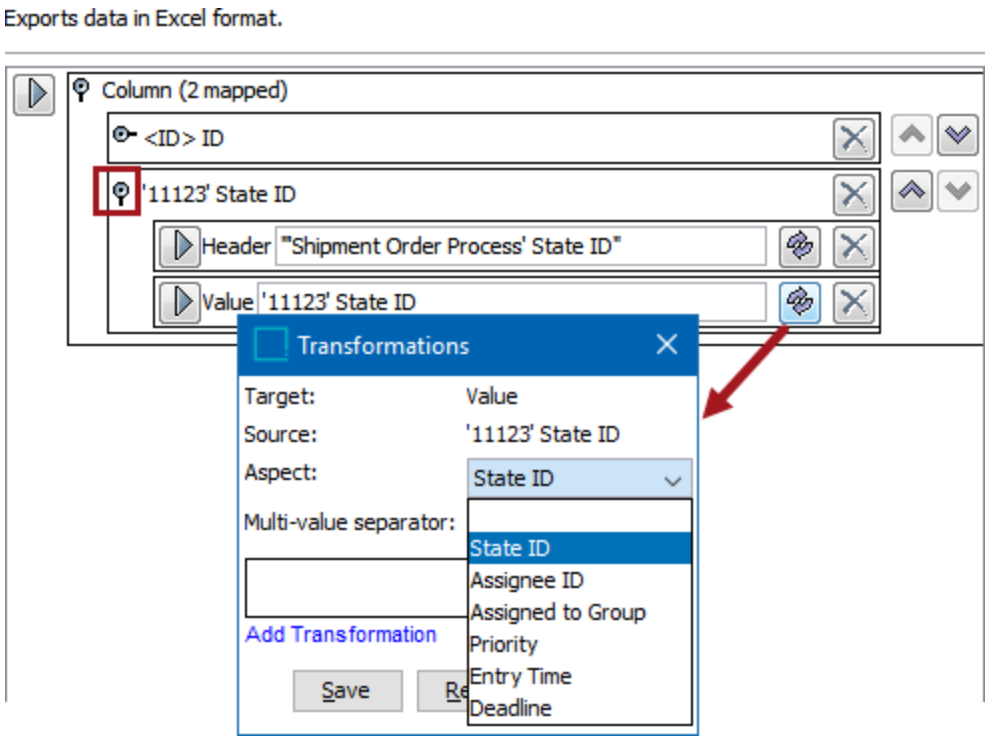
When exporting STEPXML, the option to include the information is found on the Select Format page, using the Data Objects > **Include Workflow Tasks** parameter. If set to 'yes', then each object in the exported STEPXML will have a CurrentTasks element with info similar to what is shown below.

```
<CurrentTasks>

        <Task STEPWorkflowID="Item Introduction" StateID="Technical_
Specifications" AssigneeID="Stibo Systems Technical Writers"
AssignedToGroup="true" EntryTime="2012-10-18 11:31:39" Priority="-1"/>

        <Task STEPWorkflowID="Testing 1-2-3" StateID="A_Parallel"
AssigneeID="Super user" AssignedToGroup="true" EntryTime="2012-10-31
12:30:56" Priority="-1"/>

        <Task STEPWorkflowID="Testing 1-2-3" StateID="C1" AssigneeID="Super
user" AssignedToGroup="true" EntryTime="2012-10-31 12:30:56" Priority="-
1"/>

        <Task STEPWorkflowID="Testing 1-2-3" StateID="C1-Final"
AssigneeID="Super user" AssignedToGroup="true" EntryTime="2012-11-06
12:48:50" Priority="-1"/>

        <Task STEPWorkflowID="Testing 1-2-3" StateID="C2" AssigneeID="Super
user" AssignedToGroup="true" EntryTime="2012-10-31 12:30:56" Priority="-
1"/>

        <Task STEPWorkflowID="Testing 1-2-3" StateID="C2-Initial"
AssigneeID="Super user" AssignedToGroup="true" EntryTime="2012-10-31
12:30:56" Priority="-1"/>

        <Task STEPWorkflowID="Testing 4-5-6" StateID="Some_State"
AssigneeID="Super user" AssignedToGroup="true" EntryTime="2012-11-05
14:45:18" Priority="-1"/>

</CurrentTasks>
```



The same information can be included in column based exports, but data must be mapped into columns for each workflow of interest.

To specify which type of workflow related data goes into a column, click the flipper button for the mapped data and click the transform button for the value part. For details on this setup, see the **STEP Workflow Task Info - Data Source Outbound** topic in the **Data Exchange** documentation.



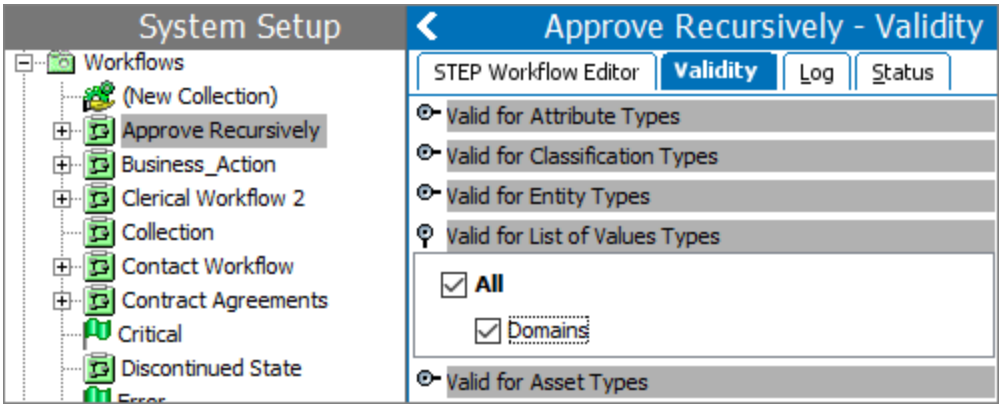
For more information about exporting in STEPXML format, see **STEPXML Outbound Parameters** in the **Data Exchange** documentation.

For more information about exporting data in Excel format, see **Excel Format** section of the **Data Exchange** documentation.

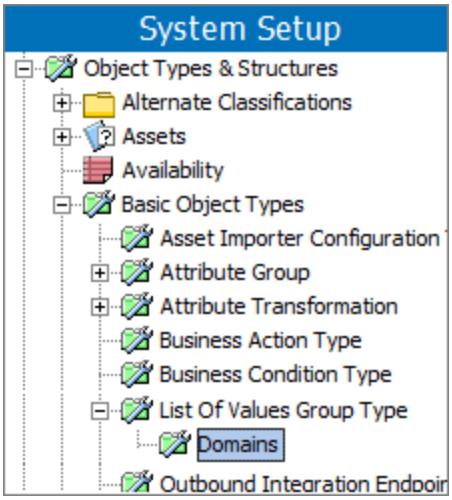
Workflows and LOVs

An LOV can be initiated into and/or removed from a workflow the same way as products, classification, entities, and assets. LOVs can be assigned to a respective user or user group to update the LOV values, similar to the enrichment of product attributes.

The workflow must be valid for the LOV object type 'Domains' as shown in the image below.



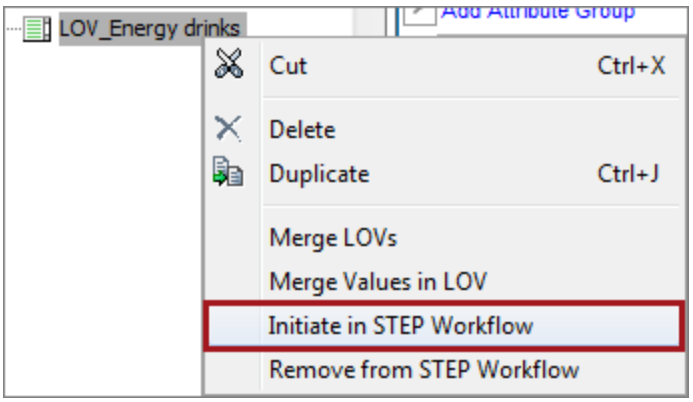
Domains are the only object type for LOVs and are found in workbench in System Setup > Object Types & Structures > Basic Object Types > List Of Values Group Type (as shown in the image below).



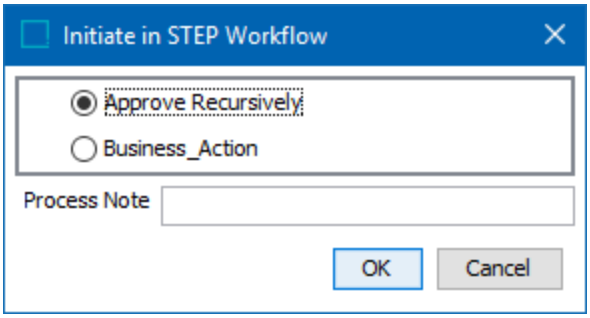
Initiate an LOV in a Workflow

Below are steps to initiate one LOV to a Workflow, however more than one LOV can be initiated to the same workflow.

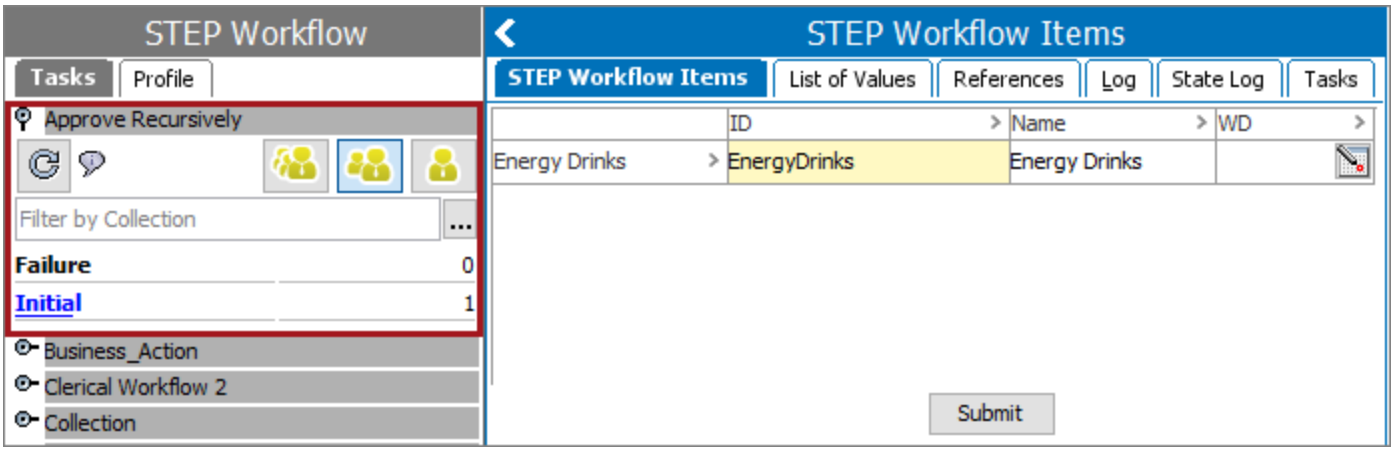
1. Right-click on an LOV, and select 'Initiate in STEP Workflow.'



2. The Initiate in STEP Workflow dialog will display listing only those workflows configured with validity for Domains (LOVs). Select the appropriate workflow, optionally enter text within the Process Note parameter, and click the **OK** button.



The LOV has now been initiated into the workflow. This can be viewed by going to STEP Workflow > Expanding the respective workflow flipper and selecting the workflow state. The workflow editor will display. Click the STEP Workflow Items tab and the LOV will display.

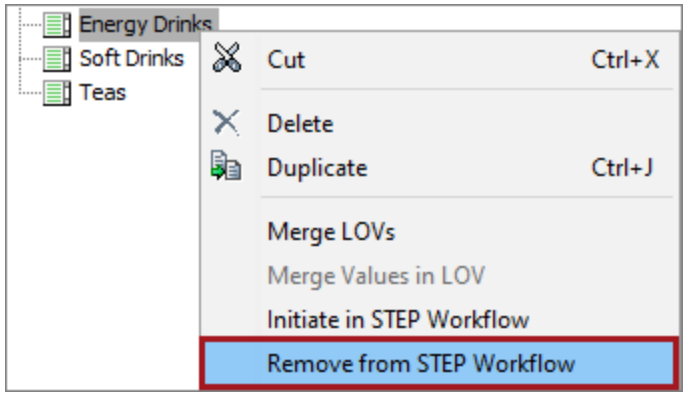


For more information, see the **Working with Tasks in Workflows** section of the **Workflows** documentation.

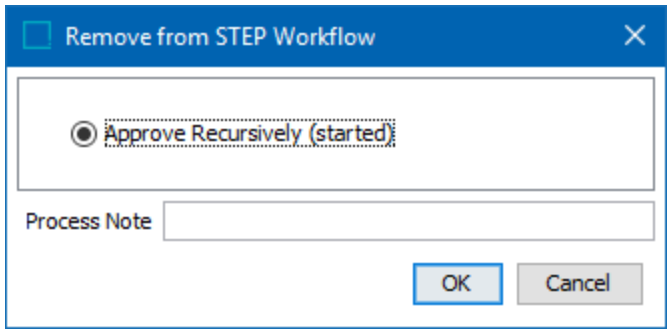
Remove an LOV from a Workflow

Below are steps to remove an LOV from a Workflow.

1. Right-click on the LOV which is already in a workflow, and select the option 'Remove from the STEP Workflow' as shown below.



2. The Remove from STEP Workflow dialog will display with the workflow in which the LOV has been initiated for. If required add text to the Process Note parameter, then click the **OK** button.



The LOV has now been removed from the workflow. For more information on workflows, see the **Workflows** guide.

Java API for Workflows

Complete documentation for API functionality related to workflows can be found in the STEP API documentation at [system]/sdk or by clicking the **STEP API Documentation** button on the STEP Start Page. This topic provides some introductory information and simple examples, but should not be considered comprehensive.

In the Public Java API "WorkflowableNode" is the main interface for Workflow related functionality. All the "workflowable" node types inherit the interface and thus you can call the available methods from the node shorthand.

If, for instance, you wanted to get the time where an object entered a specific state, on entry to the state you could have a business action with a script like the one below, storing the entry time in the script variable "entryTime".

```
var entryTime = node.getTaskByID("Workflow ID", "State ID").getEntryTime
();
```

The getTaskByID() method from the WorkflowableNode interface is called on node. This returns a Task that has the getEntryTime() method (see the Task interface).

Note that this script will only work if the object is actually in the workflow with the specified ID and in the state with the specified ID.

From a business rule used in a workflow (e.g., a local rule within the workflow itself), you can typically be pretty sure that the object is in fact in the workflow. However, if you run from a business rule used elsewhere in STEP and wanted to check whether an object was in a given flow, you could write something like:

```
if(node.getWorkflowInstanceByID("Workflow ID")) {
        // Object is in Workflow - do something
}
else {
        // Object is not in Workflow - do something else
}
```

Thus, if the object is not in the workflow, the method returns "null" - a "false" value in JavaScript.

While in the entry time example above you could be sure that you were in the workflow, likewise if you ran the script on entry to the state referenced with "State ID", you could be sure that you were in that state. However, if you are working with parallels, you could wish to do something with a task on a state that is not the same as the one you are running the business rule from. In that case it would make sense to start your script with a check for whether your object is actually in that other state:

```
var task = node.getTaskByID("Workflow ID", "ID of another State");
if(task) {
        var entryTime = task.getEntryTime();
}
else {
```

```
        logger.warning("Object with ID '" + node.getID() + "' is not in expected
State");
}
```

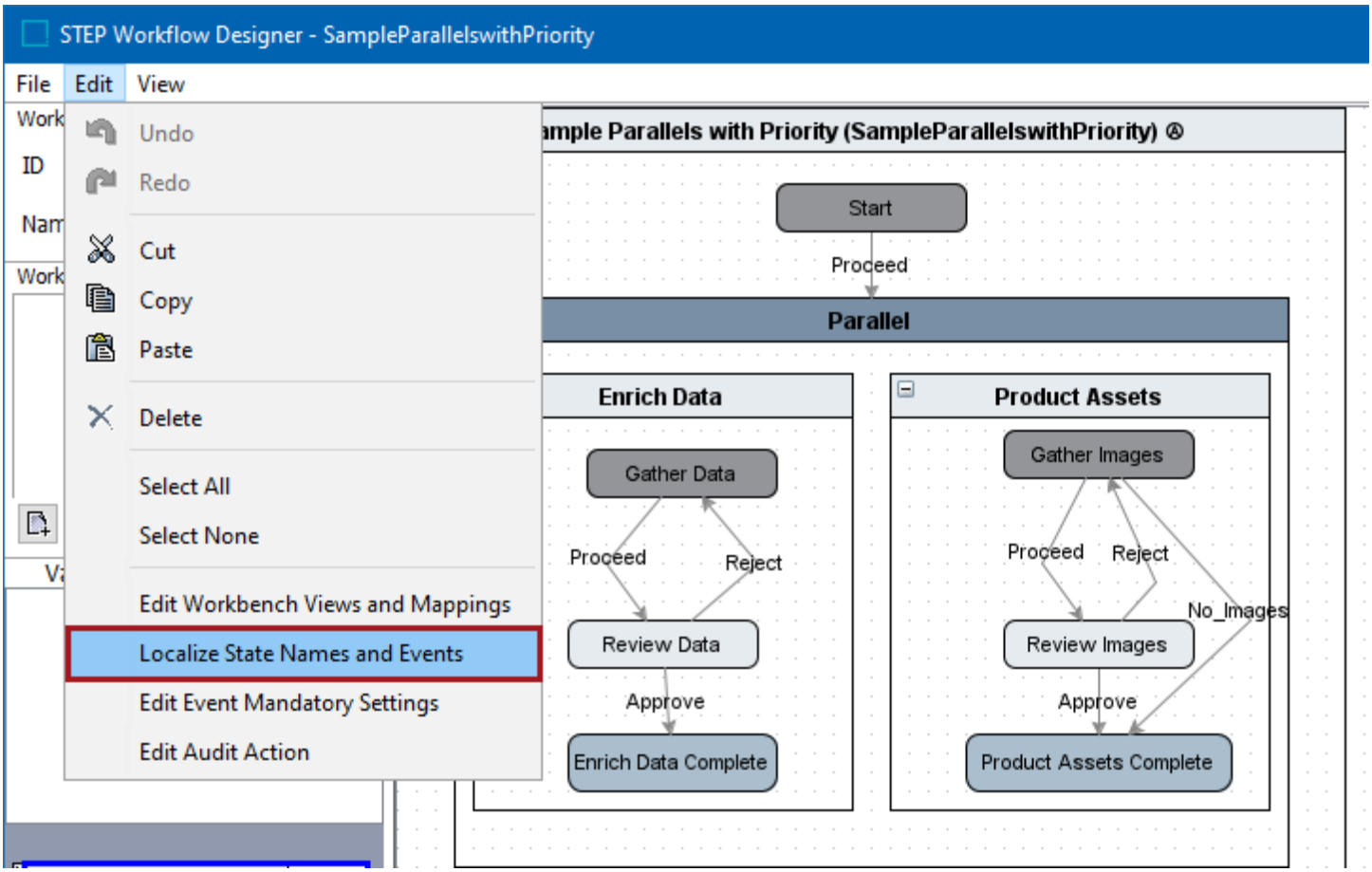
The two primary interfaces available through the WorkflowableNode interface are Task and WorkflowInstance. Via the Task interface you have methods available that relate to a given object-in-state-in-workflow relation. WorkflowInstance meanwhile gives you access to the methods relating to the object-in-workflow relation. In the Workflow Variables topic of the Advanced Workflow topics documentation, the concept of workflow variables is described. These variables live on the object-in-workflow relation and can thus be manipulated via the WorkflowInstance interface.

Localizing Workflow State Names and Events

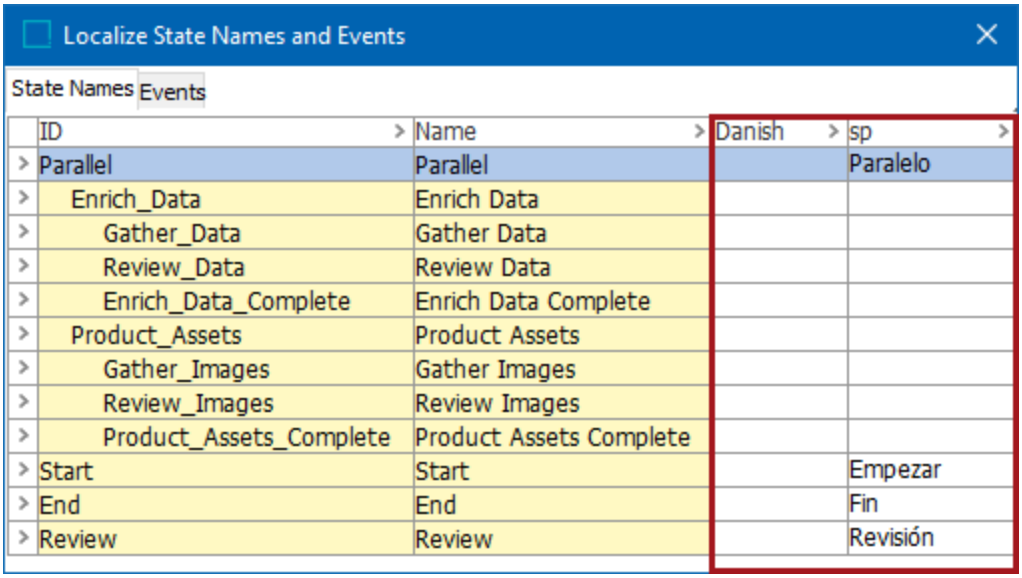
End users interact with STEP in many languages. Though a workflow may be initially created in one language, end users may use a different language and would therefore prefer to see workflow state names and events in a different language. This is possible using the Localize State Names and Events options in the STEP Workflow Designer. With this functionality, workflow designers can create a single workflow that includes localized workflow state names and events, thus eliminating the need to maintain multiple workflows for localization purposes.

To localize workflow state names or events:

1. From the System Setup tab, select the relevant workflow, right-click, and select **Edit STEP Workflow**.
2. From the Edit menu in the STEP Workflow Designer, select **Localize State Names and Events**.



3. Enter the localized values for each state name and/or event.

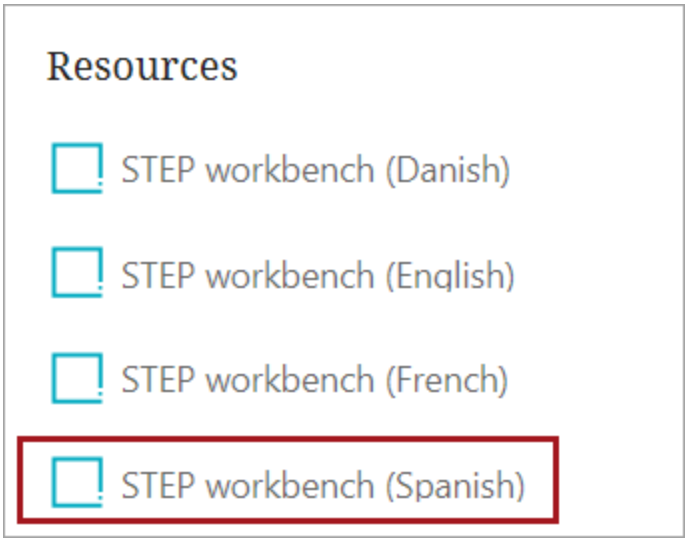


ID	Name	Danish	sp
Parallel	Parallel		Paralelo
Enrich_Data	Enrich Data		
Gather_Data	Gather Data		
Review_Data	Review Data		
Enrich_Data_Complete	Enrich Data Complete		
Product_Assets	Product Assets		
Gather_Images	Gather Images		
Review_Images	Review Images		
Product_Assets_Complete	Product Assets Complete		
Start	Start		Empezar
End	End		Fin
Review	Review		Revisión

Note: The languages available for translation are determined by two things: Translated workbench instances and the locales available in the Web UI. Web UI locales are determined by the portal messages files in the PortalLocales folder. The PortalLocales folder can be found on the application server in the location specified by the Portal.Localizer.Folder property in the sharedconfig.properties file.

Localized values entered in the workflow designer appear when users access a workflow in the relevant locale-specific Web UI or in a translated instance of the STEP Workbench. Within the workbench, this includes but is not limited to: STEP Workflow tab, object Status and State Log tabs, and dropdown search options when using a STEP Workflow search criteria.

For example, a user logging in to the Spanish workbench or Web UI locale would see the corresponding localized values entered in the workflow designer.



Note: Changing the Context selection in the workbench does not have any impact on the display of localized state names or events.

Considerations and Limitations

- The name of the workflow itself is not translatable via this feature as this is driven by context. If the context being viewed employs a different language than the translated workbench instance or selected Web UI locale, this could result in an end user seeing the name of the workflow itself and the states and/or events within the workflow in different languages.
- Offline translation of state names and events via export / import is not supported. Exports of workflows with localized state names or events include the localized values and import to new systems will reflect the localizations. If an imported workflow contains *more* locales than the target system, only locales common to both systems are displayed. If a new locale is later added on the target system and localized values exist for it via previous import, the localized values will display. If an imported workflow contains *fewer* locales than the target system, translations on the target system that are existing will be deleted, since the imported workflow definition will be seen as the governing workflow.

Mandatory Attributes and References in Workflows

Workflow mandatory attributes and references are one of the most common ways to set up mandatory attributes. The primary benefits to using this functionality are:

- Designers are given the freedom to make the right data mandatory at the right time (as opposed to using approval mandatory settings)
- Mandatory settings are configured a single time in the STEP Workflow Designer and take effect across all interfaces (workbench, Web UI, and Smartsheets), without requiring any additional configuration
- Functionality includes both visual indicators and user-facing messages at the time of enforcement (e.g., when a task is submitted from a state)

Essentially, workflow mandatory attributes provide a combination of the functionalities described above as they provide both visual indicators and enforcement, and do so across workbench, Web UI, and Smartsheets, as well as allowing for attributes to be conditionally mandatory in all of these interfaces. Because of this, workflow mandatory is the preferred method for handling mandatory attributes. However, this method is of course only available when data is being populated within a workflow, so when this is not the case, the other mandatory attribute settings described in this topic can be used.

This topic describes the functionality for making attributes, attribute groups, and references mandatory specifically within a workflow state or for a particular workflow transition.

Prerequisites

In using this functionality, it may also be helpful to understand the other mandatory attribute settings (e.g., those that are *not* specific to objects within a workflow). For more information on general mandatory attribute settings, see the **Mandatory Attributes** topic in the **Attributes** section of the **System Setup / Super User Guide**.

For mandatory **references**, there is no general functionality, other than business rules, to make references mandatory for objects that are not specifically in a workflow. However, references can be made mandatory in **Smartsheets** without the need for business rules to enforce their population. For more information on mandatory references in Smartsheets, see the **Mandatory Attributes and References in Smartsheets** topic in the **Excel Smartsheet Format** section of the **Data Exchange** documentation.

Setting up Workflow State Mandatory Data

Users can specify which **attributes**, **attribute groups**, and **references** are mandatory for a selected workflow state via the State Editor.

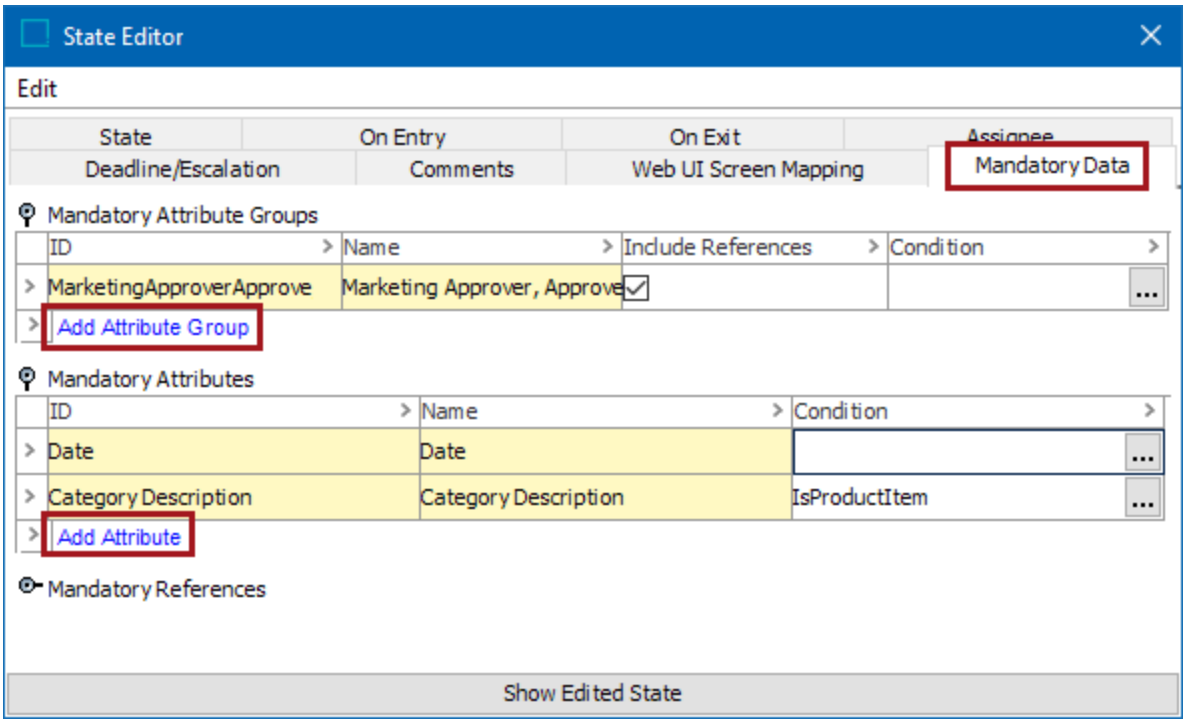
Setting up Workflow State Mandatory Attributes

The following steps describe how to specify which attributes and attribute groups are mandatory for a workflow state via the State Editor.

Mandatory attributes configured on a state will be marked with an asterisk when working in Web UI. Mandatory attributes configured on a state will be marked with an asterisk in the *workbench* **only** if the 'Mandatory' checkbox is selected when adding the attribute to the workbench view. For more information, see the **Configuring Views for Workflows** topic in the **Workflows** documentation.

Note: No markers are displayed to indicate mandatory attributes for workflow *transitions*.

1. In **System Setup**, locate and select the desired workflow.
2. Right-click and select **Edit STEP Workflow**.
3. In the STEP Workflow Designer, select the relevant state, then right-click and select **Edit State** to edit the state.
4. Select the **Mandatory Data** tab in the State Editor.



5. Click **Add Attribute Group** or **Add Attribute** to add an attribute or attribute group to the list of mandatory attributes.
6. If the mandatory status of an attribute or attribute group needs to be determined by a business condition, click the ellipsis button (...) next to the applicable attribute or attribute group. In the Mandatory Condition editor, configure an appropriate business condition or select a global business rule.

Note: Unless the condition returns 'False,' the attribute will remain mandatory.

For more information about conditionally mandatory attributes, see the **Conditionally Mandatory Attributes and References in Workflows** section of the **Workflows** documentation.

7. **Save** the workflow.

If any mandatory attributes are missing when submitting an object in workbench or Web UI, an error message listing all missing mandatory attributes will appear, preventing the user from progressing through the workflow. For more information on how Web UI handles this, see the **Workflow Mandatory Attributes in Web UI** topic in the **Web UI Getting Started** documentation.

In Excel Smartsheets, users will receive an error message when validating a sheet with missing mandatory attributes. For more information, see the **Mandatory Attributes and References in Smartsheets** topic in the **Excel Smartsheet Format** section of the **Data Exchange** / **Data Formats** documentation.

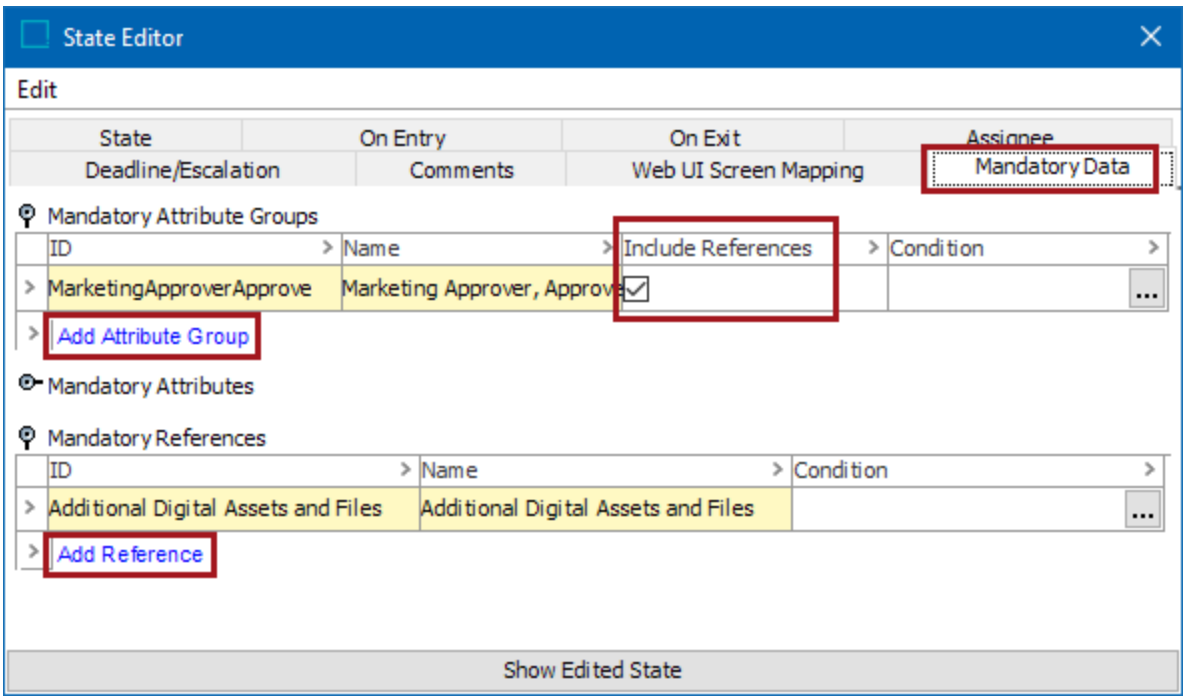
Setting up Workflow State Mandatory References

Mandatory references are set up in a similar way to mandatory attributes in the State Editor. Mandatory references configured on a state will be marked with an asterisk in the following Web UI components:

- References
- Reference Value
- Reference Header
- Referenced Asset Representation
- Classification Links
- Classification Product Links

Note: No markers are displayed to indicate mandatory references for workflow *transitions*.

1. In **System Setup**, locate and select the desired workflow.
2. Right-click and select **Edit STEP Workflow**.
3. In the STEP Workflow Designer, select the relevant state, then right-click and select **Edit State** to edit the state.
4. Select the **Mandatory Data** tab in the State Editor.



5. Click **Add Attribute Group**, then check the **Include References** box to make all references contained in the attribute group mandatory for the workflow state.
6. Click **Add Reference** to add a reference to the list of mandatory references.
7. If the mandatory status of a reference or references within an attribute group need to be determined by a business condition, click the ellipsis button (...) next to the applicable reference or attribute group. In the Mandatory Condition editor, configure an appropriate business condition or select a global business rule.

   Note: Unless the condition returns 'False,' the reference will remain mandatory.

   For more information about conditionally mandatory references, see the **Conditionally Mandatory Attributes and References in Workflows** section of the **Workflows** documentation.
8. **Save** the workflow.

If any mandatory references are missing when submitting an object in workbench or Web UI, an error message listing all missing mandatory references will appear, preventing the user from progressing through the workflow.

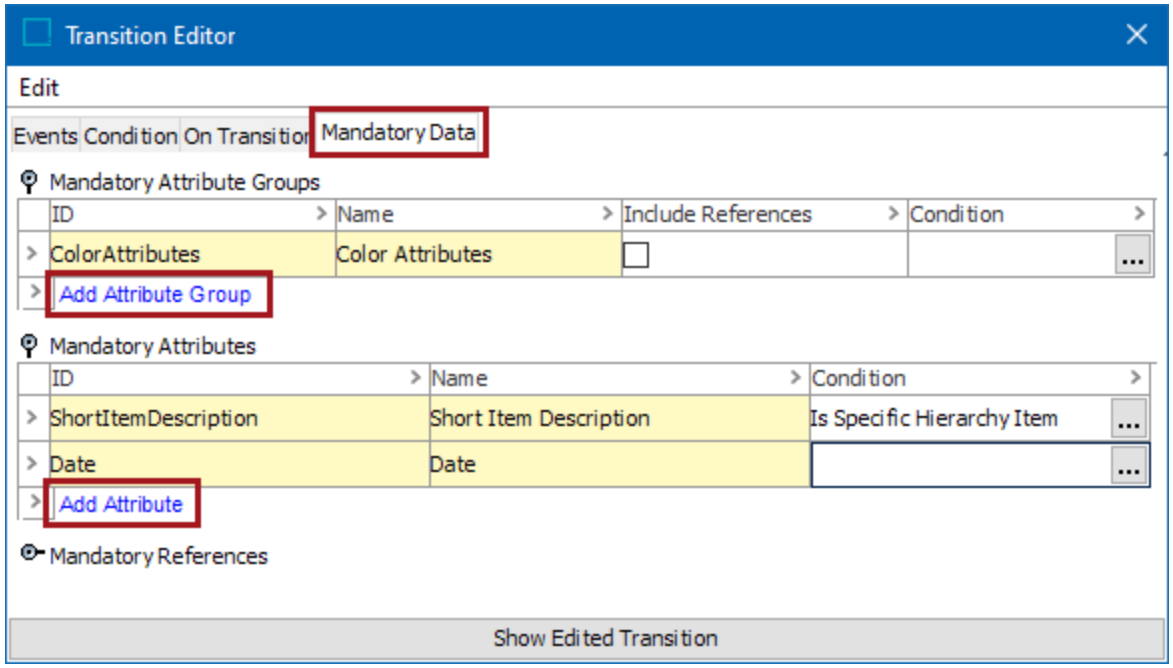
In Excel Smartsheets, users will receive an error message when validating a sheet with missing mandatory references. For more information, see the **Mandatory Attributes and References in Smartsheets** topic in the **Excel Smartsheet Format** section of the **Data Exchange** / **Data Formats** documentation.

Setting up Workflow Transition Mandatory Data

Users can specify which **attributes**, **attribute groups**, and **references** are mandatory on workflow transitions via the Transition Editor.

Setting up Workflow Transition Mandatory Attributes

1. In **System Setup**, locate and select the desired workflow.
2. Right-click and select **Edit STEP Workflow**.
3. In the STEP Workflow Designer, select the relevant transition, then right-click and select **Edit Transition** to edit the transition.
4. Select the **Mandatory Data** tab in the **Transition Editor**.



5. Click **Add Attribute Group** or **Add Attribute** to add an attribute group / attribute to the list of mandatory attributes.
6. If the mandatory status of an attribute needs to be determined by a business condition, click the ellipsis button (...) next to the applicable attribute. In the Mandatory Condition editor, configure an appropriate business condition or select a global business rule.

   Note: Unless the condition returns 'False,' the attribute will remain mandatory.

   For more information about conditionally mandatory attributes, see the **Conditionally Mandatory Attributes and References** section of the **Workflows** documentation.

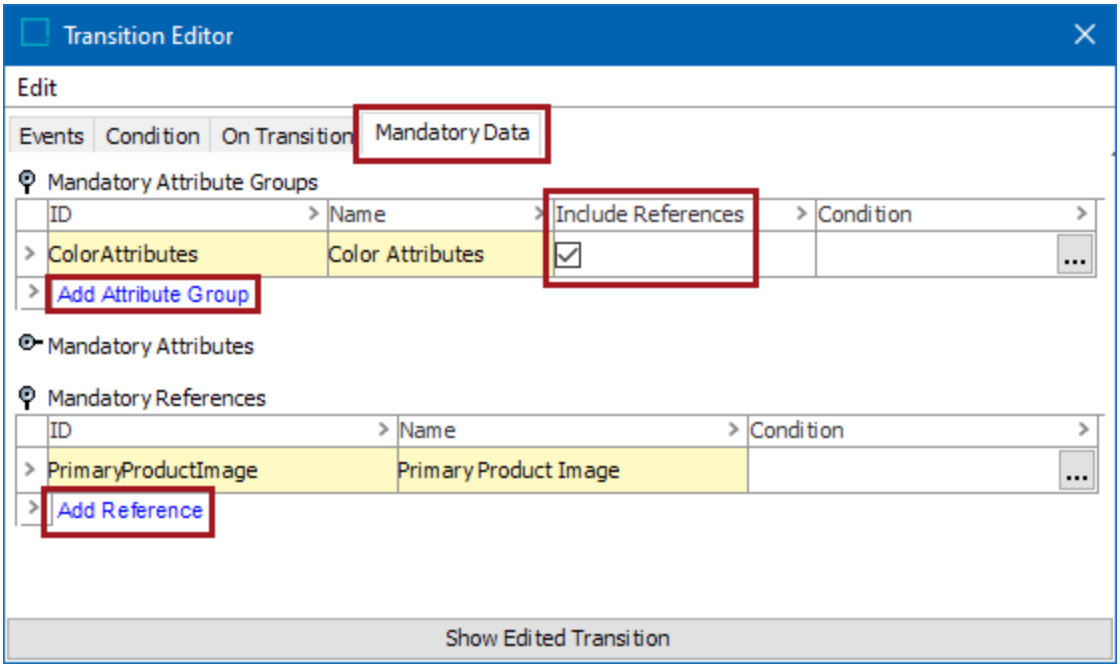
7. **Save** the workflow.

If any mandatory attributes are missing when submitting an object in workbench or Web UI, an error message listing all missing mandatory attributes will appear, preventing the user from progressing through the workflow. For more information on how Web UI handles this, see the **Workflow Mandatory Attributes in Web UI** section of the **Web User Interfaces / Workflows in Web UI** documentation.

Setting up Workflow Transition Mandatory References

Mandatory references are set up in a similar way to mandatory attributes in the Transition Editor.

1. In **System Setup**, locate and select the desired workflow.
2. Right-click and select **Edit STEP Workflow**.
3. In the STEP Workflow Designer, select the relevant transition, then right-click and select **Edit Transition** to edit the transition.
4. Select the **Mandatory Data** tab in the **Transition Editor**.



5. Click **Add Attribute Group**, then check the **Include References** box to make all references contained in the attribute group mandatory for the workflow transition.
6. Click **Add Reference** to add a reference to the list of mandatory references.
7. If the mandatory status of a reference needs to be determined by a business condition, click the ellipsis button (...) next to the applicable attribute group or reference. In the Mandatory Condition editor, configure an appropriate business condition or select a global business rule.

Smartsheets State Mandatory Attributes and References

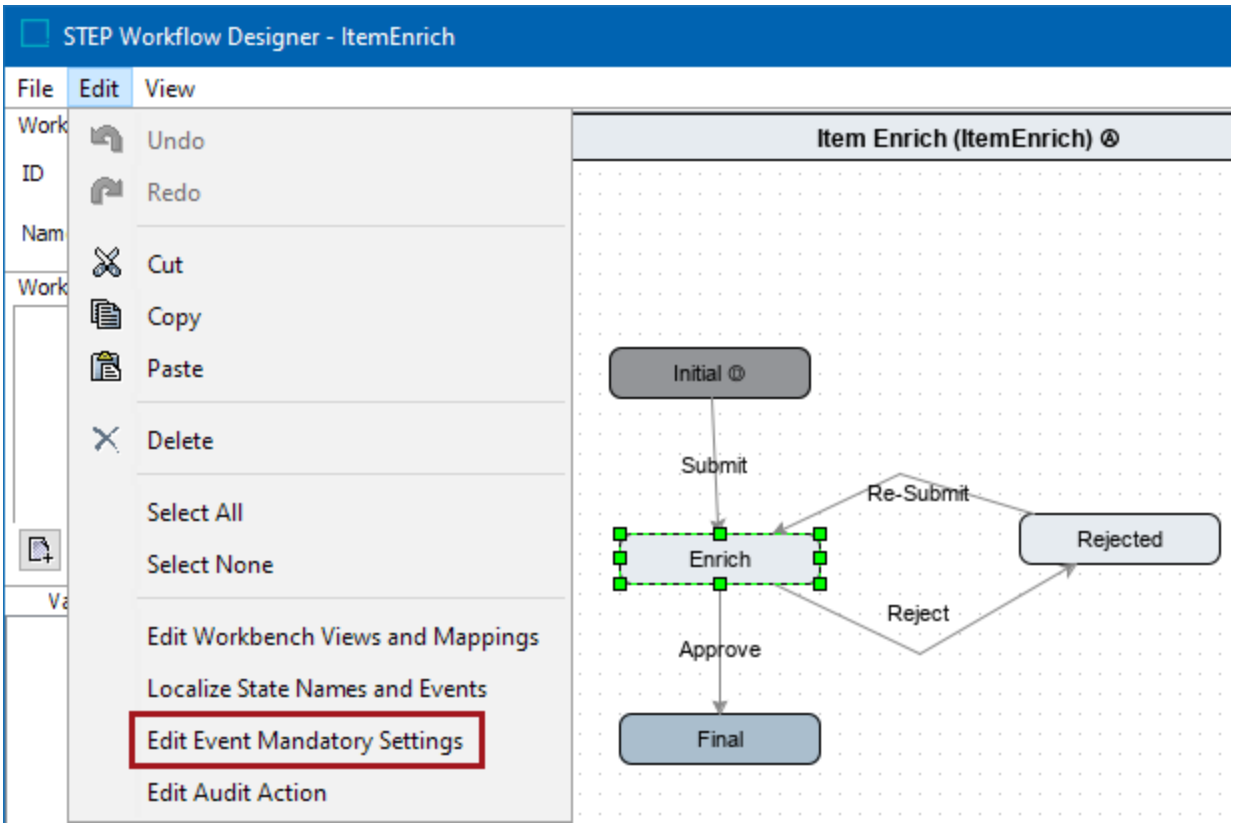
In Smartsheets, it is possible to enforce the mandatory status of attributes, attribute groups, and/or references on workflow **states**. For more information, see the **Mandatory Attributes and References in Smartsheets** topic in the **Excel Smartsheet Format** section of the **Data Exchange** / **Data Formats** documentation.

Set Workflow to Ignore Mandatory Attributes and References

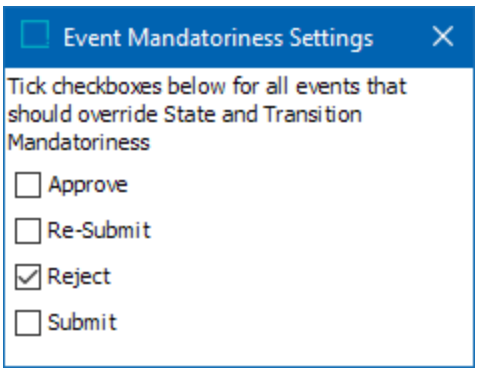
After setting mandatory attributes and/or references on a state, transition, or on a Web UI screen, specific events can be identified for which the population of the attributes and/or references should not be enforced. This allows an object to move through designated states in a workflow, even when the object is missing mandatory attribute values or linked references. The following example uses mandatory attributes, though the functionality behaves identically for mandatory references.

In the below example, 'Color' and 'Material' are set as mandatory attributes on the **Enrich** state. If the user does not populate these values on a given product and attempts to submit from the state via any transition (**Reject** or **Approve**), the object would not be able to proceed through the workflow until the missing values were provided. However, if the **Reject** event is set to ignore mandatory attributes, the user may send the product to the 'Rejected' state even if 'Color' and 'Material' have missing values.

1. In the workflow editor, navigate to Edit > **Edit Event Mandatory Settings**.



2. In the 'Event Mandatoriness Settings' pop-up window, check the appropriate event(s) for which mandatory attribute and reference requirements should be ignored, then close the window.



3. **Save** the workflow.

Note: While this can also be accomplished by setting mandatory attributes and references on transitions themselves, the Web UI only provides a visual indicator (asterisk) on mandatory attributes and references that are set in workflow *states*. This feature allows users to see the visual indicator for mandatory attributes and references required for forward progression, while also being able to disregard these requirements for certain transitions, such as Rejections.

Mandatory attributes set on a Web UI screen are only honored if the 'Submit Action' has 'Enforced Validity' checked. For more information, see the **Mandatory Attributes in Web UI** topic in the **Web User Interfaces / Web UI Getting Started** documentation.

Conditionally Mandatory Attributes and References in Workflows

Sometimes it is essential for an attribute or reference to be populated in one or two specific cases, but is otherwise unnecessary. To account for these cases, the mandatory status of attributes and references on objects moving through a workflow can be enforced via **business conditions** set on a state or transition.

Mandatory attributes and references configured with a condition will remain mandatory unless that condition returns false, indicating that the condition was not met and the attributes in question are not required when proceeding to the next state in the workflow. Like standard mandatory attributes and references, if conditionally mandatory attributes / references are missing values when attempting to submit an object to the next state, the object is prevented from progressing through the workflow and the user receives a message listing the attributes / references that have missing mandatory values. In Web UI, these conditions can be evaluated on the fly and dynamically reflected as the user works with the data, with a change in mandatory status being reflected on the screen in real time, allowing users to react immediately, without having to submit and receive an error message.

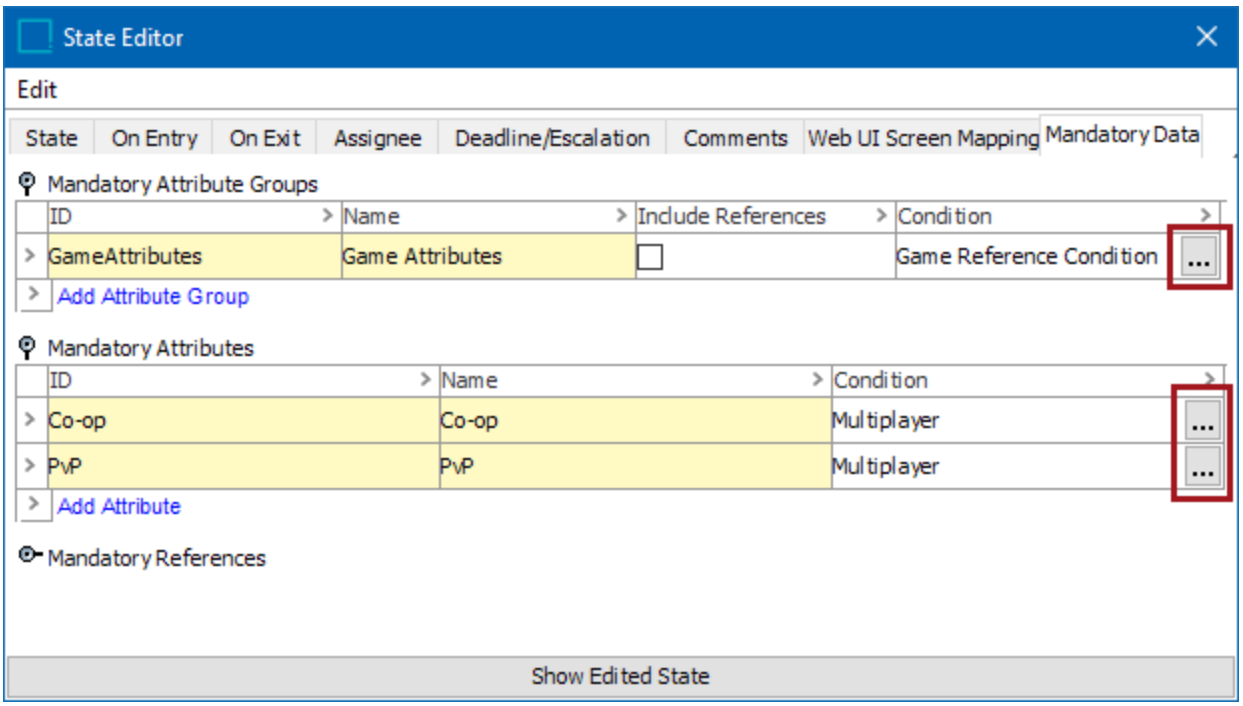
For more information about setting mandatory attributes and references in workflows, see the **Mandatory Attributes and References in Workflows** section of the **Workflows** documentation.

For more information about business conditions, see the **Business Conditions** section of the **Business Rules** documentation.

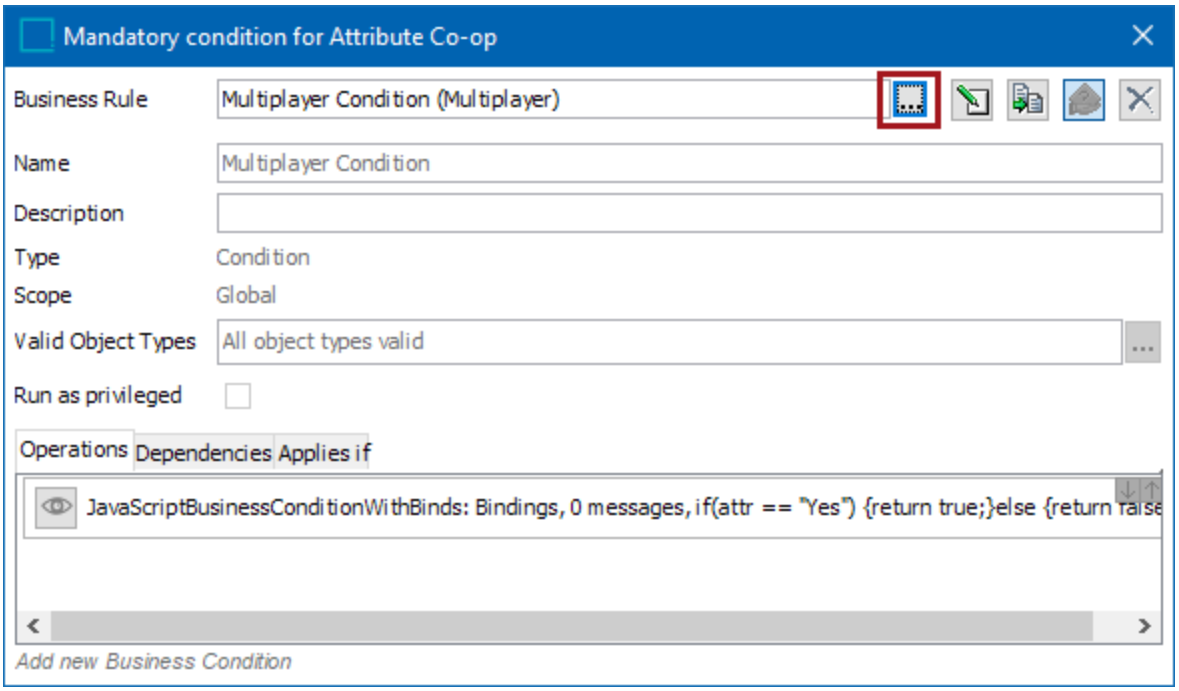
Configuring Workflow States / Transitions with Conditionally Mandatory Attributes

A condition can be mapped to an **attribute** or a **group of attributes** on either a state or transition in a workflow.

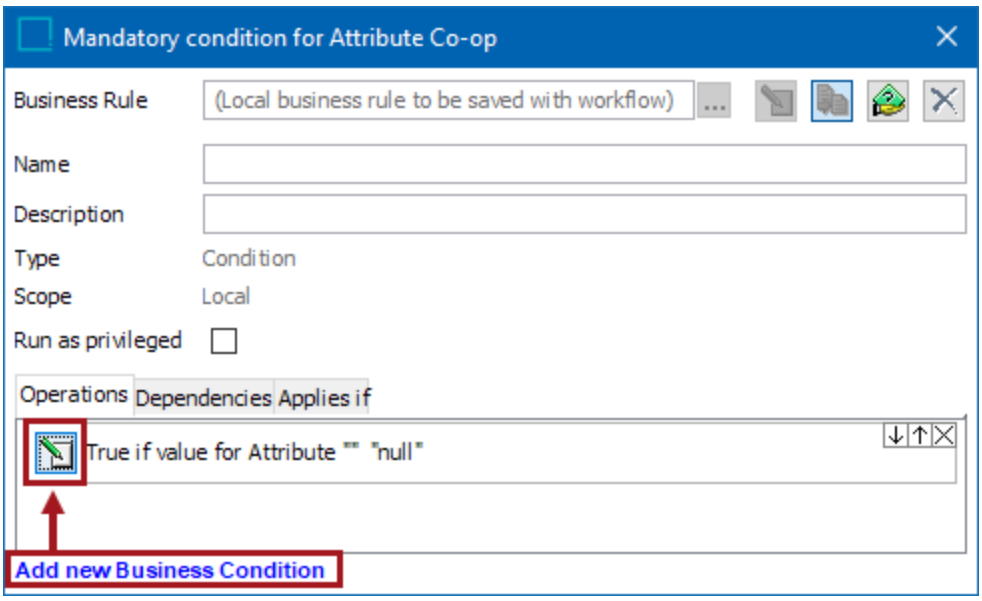
1. Navigate to the **Mandatory Data** tab in the State / Transition Editor.
2. Locate the relevant attribute group and/or attribute(s) and click the ellipsis button (...) in the 'Condition' column.



3. In the 'Mandatory condition for' editor that displays, specify the condition to use by selecting a global business rule via the 'Business Rule' ellipsis button (...). Once the selection is made, close the editor.



Alternatively, you can create a new business condition by clicking the 'Add new Business Condition' link at the bottom of the editor, and then click the **Edit Operation** button that appears. From the dropdown, select 'Evaluate JavaScript,' then enter the business condition in the JavaScript area.



Important: For use with conditionally mandatory attributes, the business condition should only return 'True' or 'False.' If anything else is returned it will be treated as 'True'.

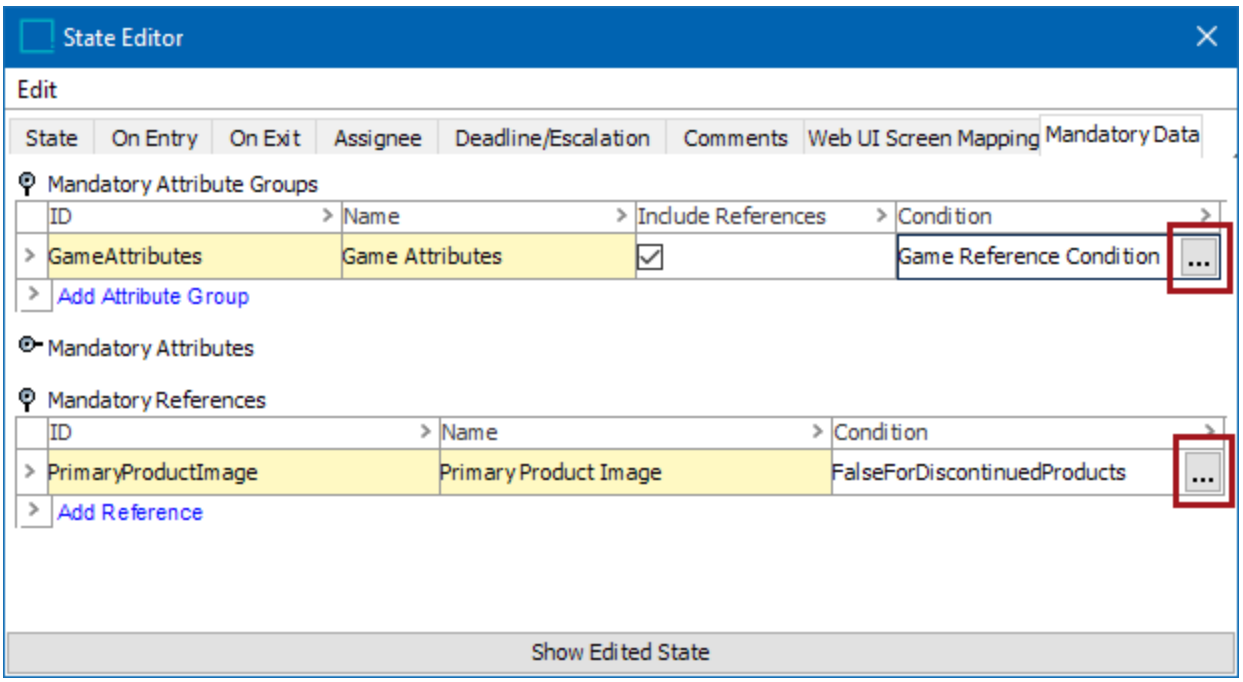
4. Once finished, click **Save** and close the editor.
5. Close the State / Transition Editor and Save the workflow.

Configuring Workflow States / Transitions with Conditionally Mandatory References

Conditionally mandatory **references** on workflow states and transitions are configured similarly to conditionally mandatory attributes. Conditions can be mapped to all references listed within an attribute group or to individual references.

1. Navigate to the **Mandatory Data** tab in the State / Transition Editor.
2. Locate the relevant attribute group and/or reference and click the ellipsis button (...) in the 'Condition' column.

Note: To apply a condition to all references within an attribute group, the 'Include References' box must first be checked. Also note that the condition will apply to both the attributes and the references in the group, so the condition should be written to take this into consideration.



3. In the 'Mandatory condition for' editor that displays, specify the condition to use by selecting a global business rule via the 'Business Rule' ellipsis button (...). Once the selection is made, close the editor. (This editor is identical to the one that displays for conditionally mandatory attributes.)

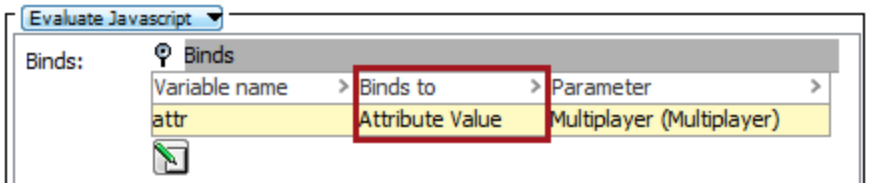
   Alternatively, you can create a new business condition by clicking the 'Add new Business Condition' link at the bottom of the editor, and then click the **Edit Operation** button that appears. From the dropdown, select 'Evaluate JavaScript,' then enter the business condition in the JavaScript area.

   Important: For use with conditionally mandatory references, the business condition should only return 'True' or 'False.' If anything else is returned it will be treated as 'True'.

4. Once finished, click **Save** and close the editor.
5. Close the State / Transition Editor and Save the workflow.

Conditionally Mandatory Attribute and Reference Handling in Web UI

In **Web UI**, the mandatory status of attributes and references can be evaluated on the fly as the user works with the data, with any change in mandatory status being reflected on the screen in real time. In order to display these dynamic changes, the attribute values driving the condition must be bound to that condition in the business rule configuration. While the condition driving the mandatory status of an attribute or reference can take more than just attributes into account, if something else is driving the condition, such as a reference, the page will need to be refreshed to update the interface.



Important: The use of business rules to determine mandatory status of an attribute or reference should be done with care, as the performance of the dynamic behavior of the editors could be impacted if highly complex business rules are used.

When viewing a state in a Node Editor, any attributes or references that are currently mandatory are marked with an asterisk. If the condition returns false due to changes in data on the page, these indicators vanish. Additionally, if there are any missing attributes that are currently mandatory, the Submit / Approve button will be locked.

Note: Conditionally mandatory attributes and references on transitions do not have visual indicators to end users since they are not evaluated until the transition is selected. That said, the warning message will still appear listing the missing attribute values / references.

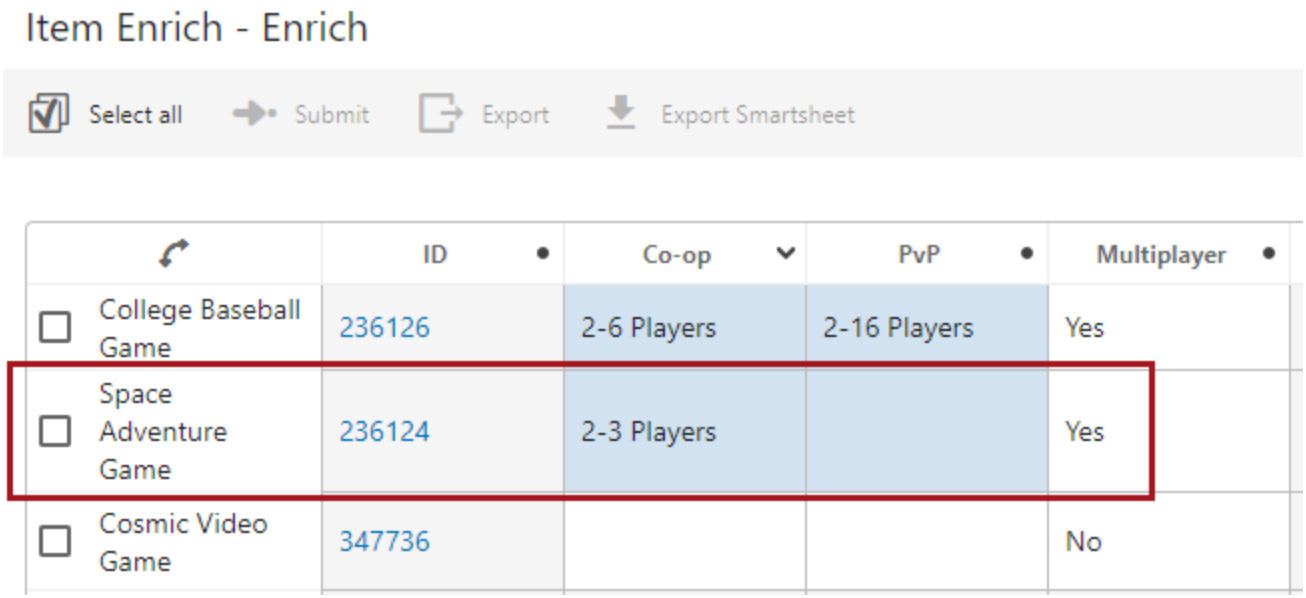
On Node Lists using Multi-Edit Display Mode, conditionally mandatory attributes and references are indicated by highlighting the cell blue. As with the Node Editor screen, the mandatory status is dynamically evaluated so if the condition returns false, the highlighting disappears.

Example

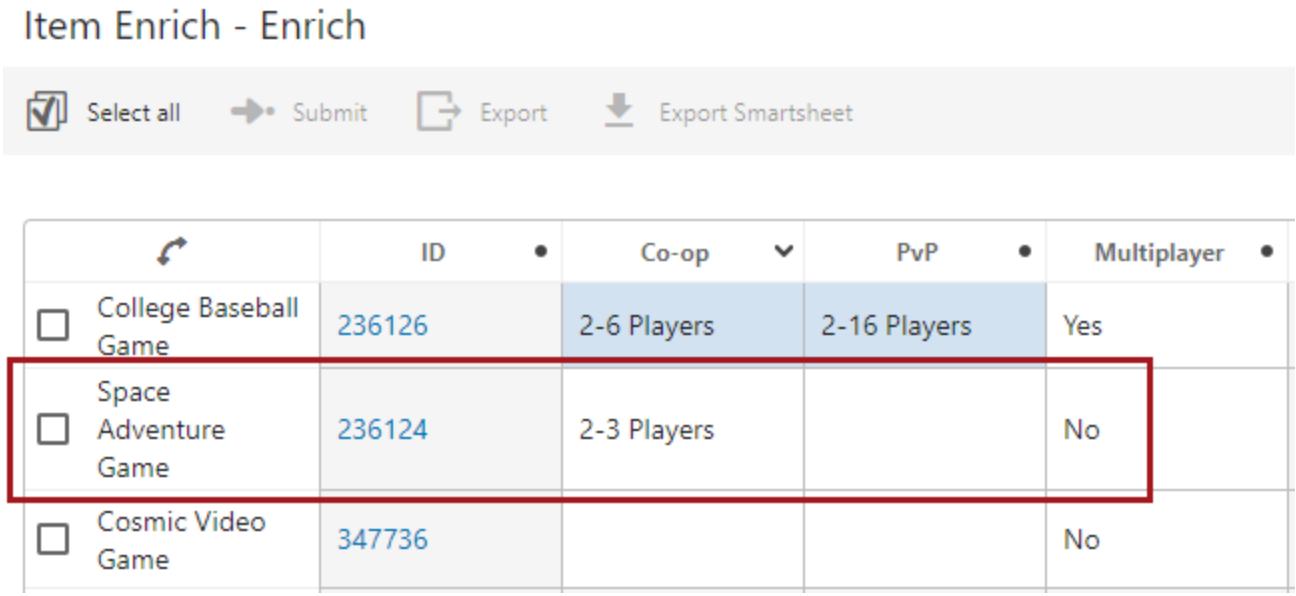
Note: This example uses conditionally mandatory attributes to describe the functionality, but the behavior is the same for conditionally mandatory references.

The business condition states that if the value of the attribute 'Multiplayer' is 'Yes', the attributes 'Co-op' and 'PvP' are mandatory. Otherwise, 'Co-op' and 'PvP' are not mandatory.

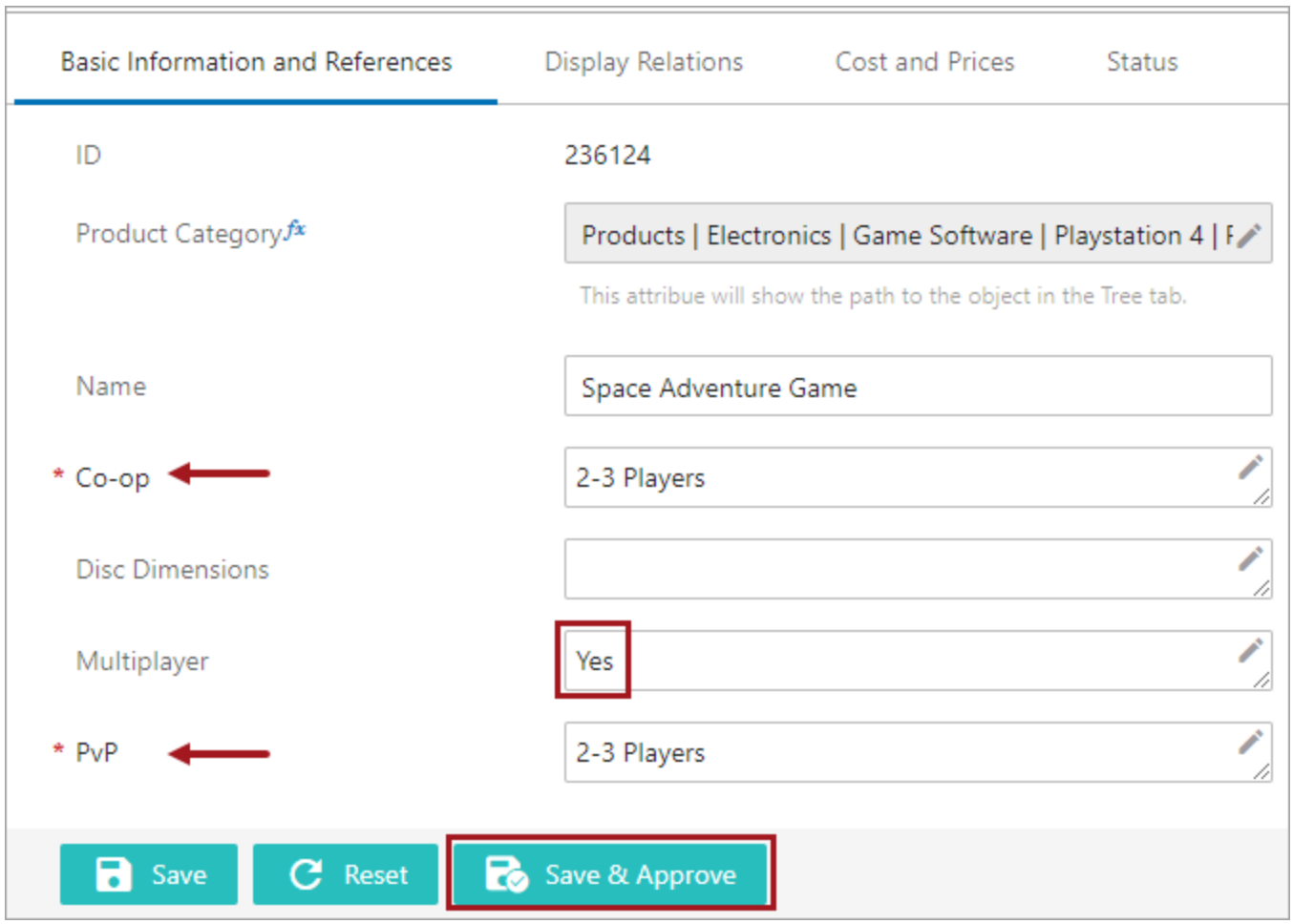
In the image below, the condition is returning true for the Space Adventure Game, so 'Co-op' and 'PvP' are mandatory (and are highlighted blue). Clicking Submit / Approve if either mandatory value is missing will result in an error message listing which mandatory attribute(s) are missing. If both attributes have values they are free to move on to the next state.



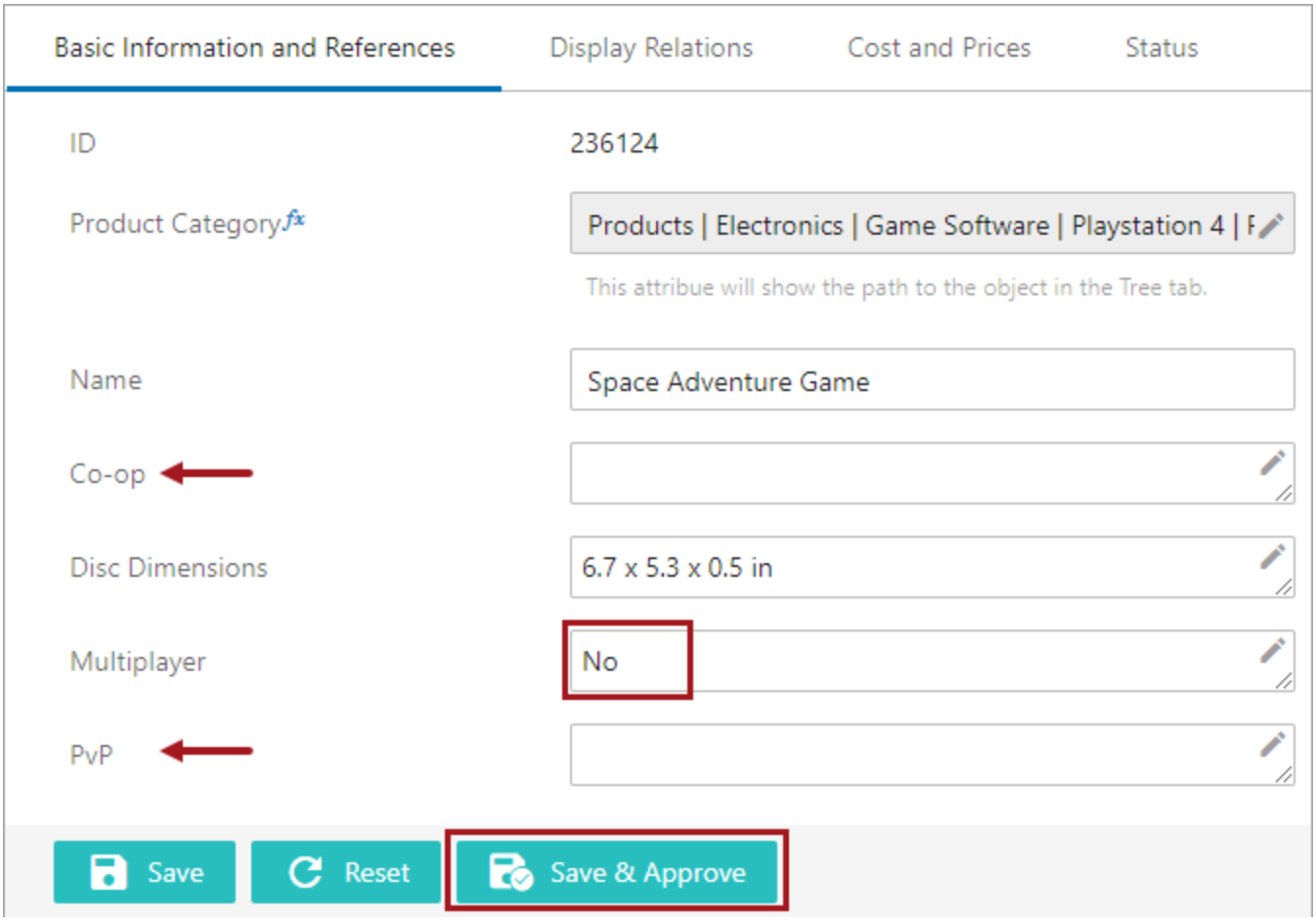
In the image below, the condition is returning false for the Space Adventure Game, so 'Co-op' and 'PvP' are not mandatory (and are not highlighted). This object is free to move on to the next state, even if 'Co-op' and 'PvP' don't have values.



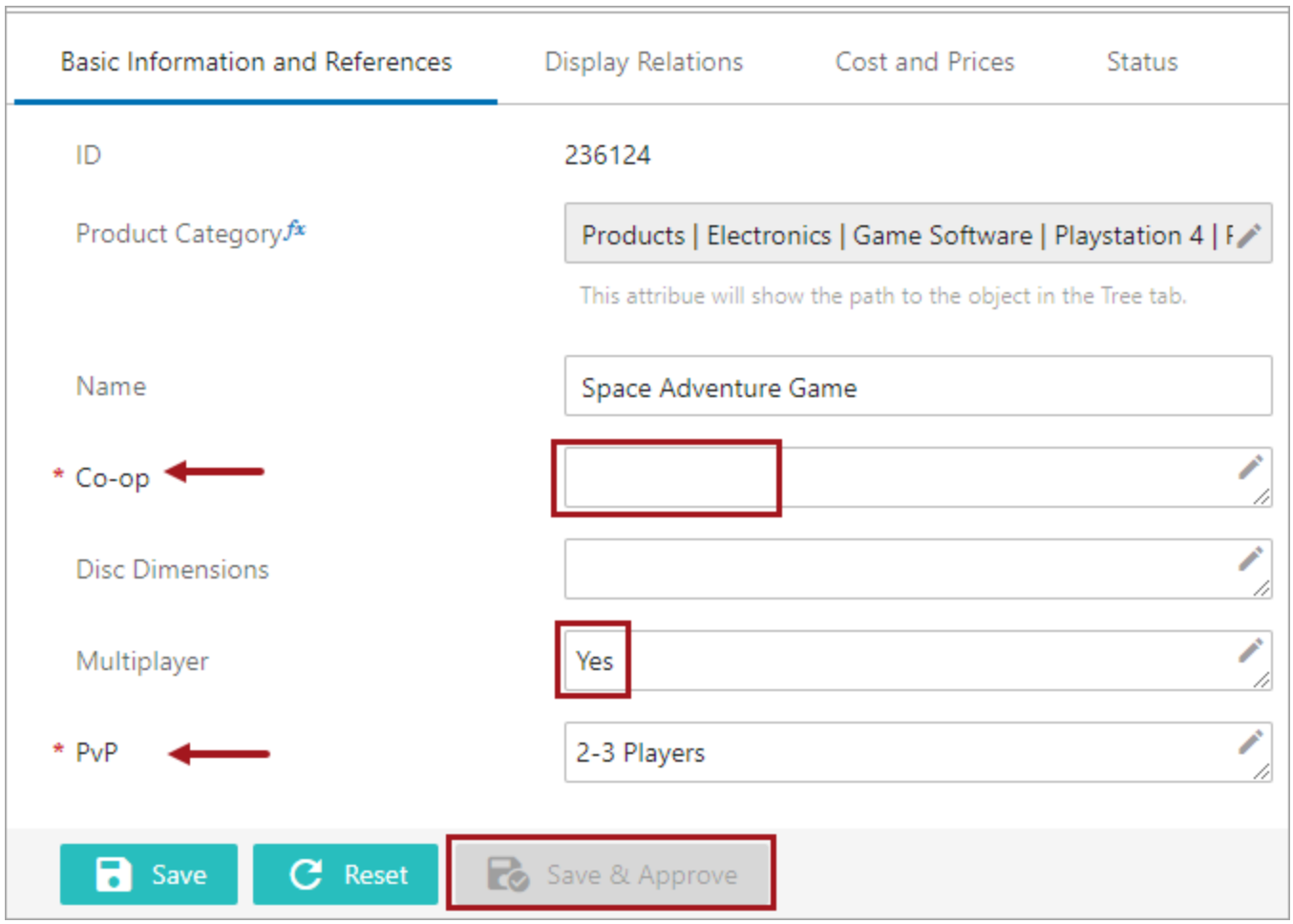
In the image below, the condition is returning true, so 'Co-op' and 'PvP' are mandatory (and are marked with asterisks). Since they have values, the Submit / Approve button is enabled.



In the image below, the condition is returning false, so 'Co-op' and 'PvP' are not mandatory (and are not marked with asterisks). Since it does not matter if they have values, the Submit / Approve button is enabled.



In the image below, the condition is returning true, so 'Co-op' and 'PvP' are mandatory (and are marked with asterisks). Since 'Co-op' is missing a value, the Submit / Approve button is disabled.



Conditionally Mandatory Attribute and Reference Handling in Workbench

Support for conditionally mandatory attributes and references is also available in the workbench, though visual indicators are not managed in relation to this. When attempting to submit an object, conditionally mandatory attributes / references are evaluated and an error is presented to the user listing any missing conditionally mandatory attributes / references (in the same manner that standard mandatory attributes / references are reported).

Determining Mandatory Status of an Attribute or Reference

Any attributes / attribute groups and/or references specified under the 'Mandatory Data' tab of a workflow state or transition are considered mandatory if:

- There is no condition set for the attribute / attribute group or reference(s)
- The condition is not applicable for the object type of the object being evaluated
- There is a pre-condition on the rule that is not met and therefore the condition is not evaluated

- The condition fails
- The condition returns 'True'

The only case in which an attribute / attribute group or reference is *not* considered mandatory is if the condition returns 'False'.

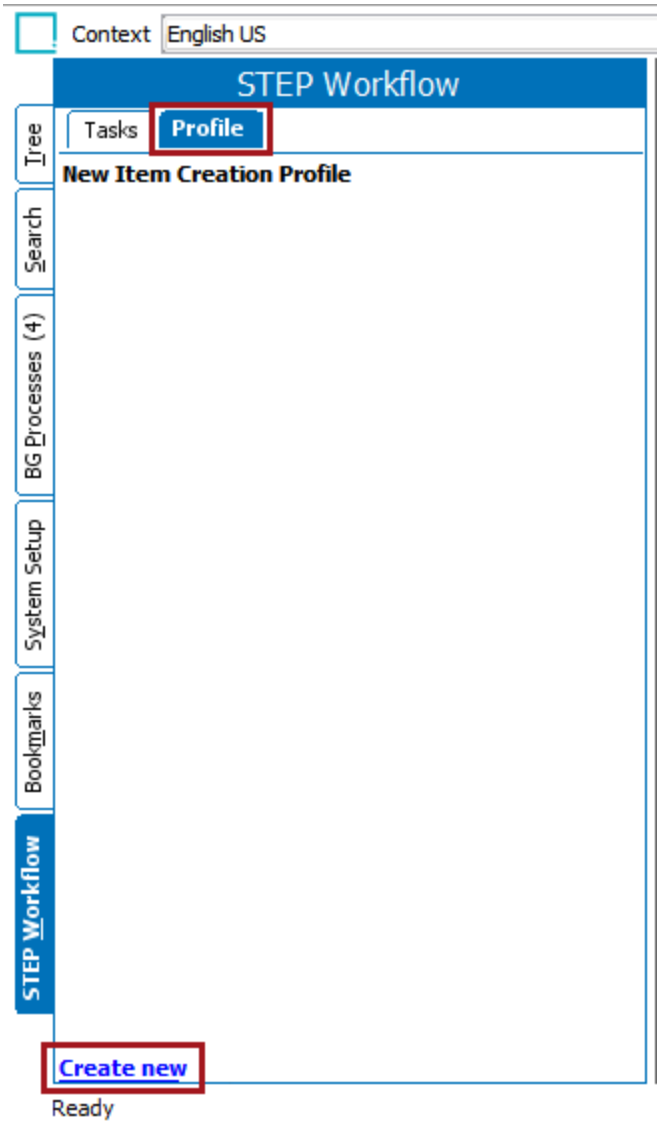
Because the mandatory status of an individual attribute or reference can be determined in a number of different ways, some methods must take priority over others:

- An individual attribute or reference specified on the state / transition takes priority over attributes / references specified as part of a group.
- If an attribute / reference is listed via multiple groups, the group configured with a condition takes priority.
- If an attribute / reference is listed via multiple groups and more than one of those groups is configured with a condition, only the first condition listed is applied.
- If the attribute is explicitly defined as mandatory in Web UI via being set as mandatory on the screen, that setting overrides any conditions that may say otherwise. (This does not apply to references, which cannot be set as anything other than 'visually mandatory' on Web UI screens)

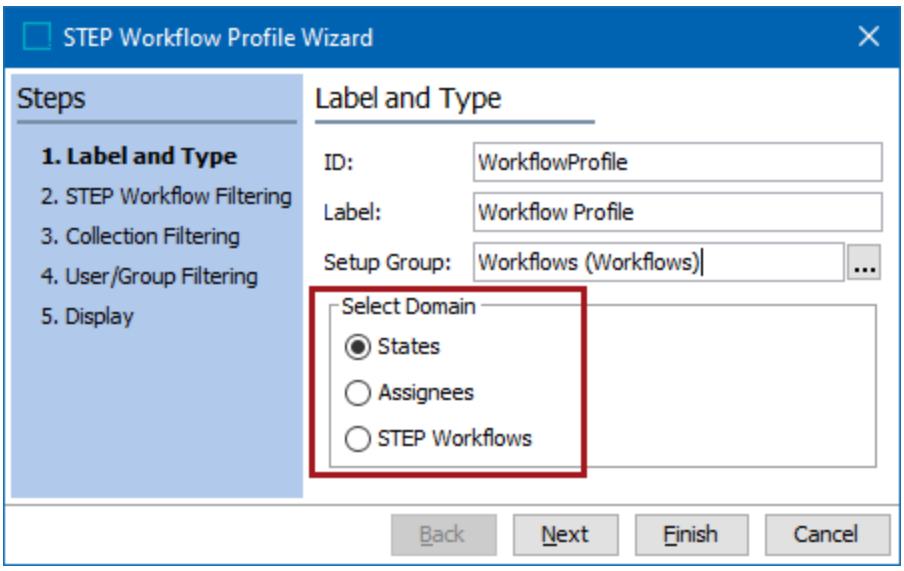
Monitoring Workflows

Workflows can be monitored via Workflow Profiles. Profiles are maintained on the Profile sub-tab of the STEP Workflow navigator tab. Profiles are available to users with the 'View and Maintain STEP Workflow Profile' privilege. A Profile can contain a wealth of information about workflows, such as: number of exceeded deadlines, how long the tasks have been assigned to the assignees, what is the workflow throughput this month versus the last six months, etc. This is all configured using the STEP Workflow Profile Wizard.

1. To create a new Profile, navigate to the STEP Workflow tab, select the Profile sub-tab, and click the **Create new** link at the bottom of the Profile sub-tab.



1. For Label and Type, provide an ID, Label (name), assign a Setup Group, and choose a Domain (from the options in the following table).



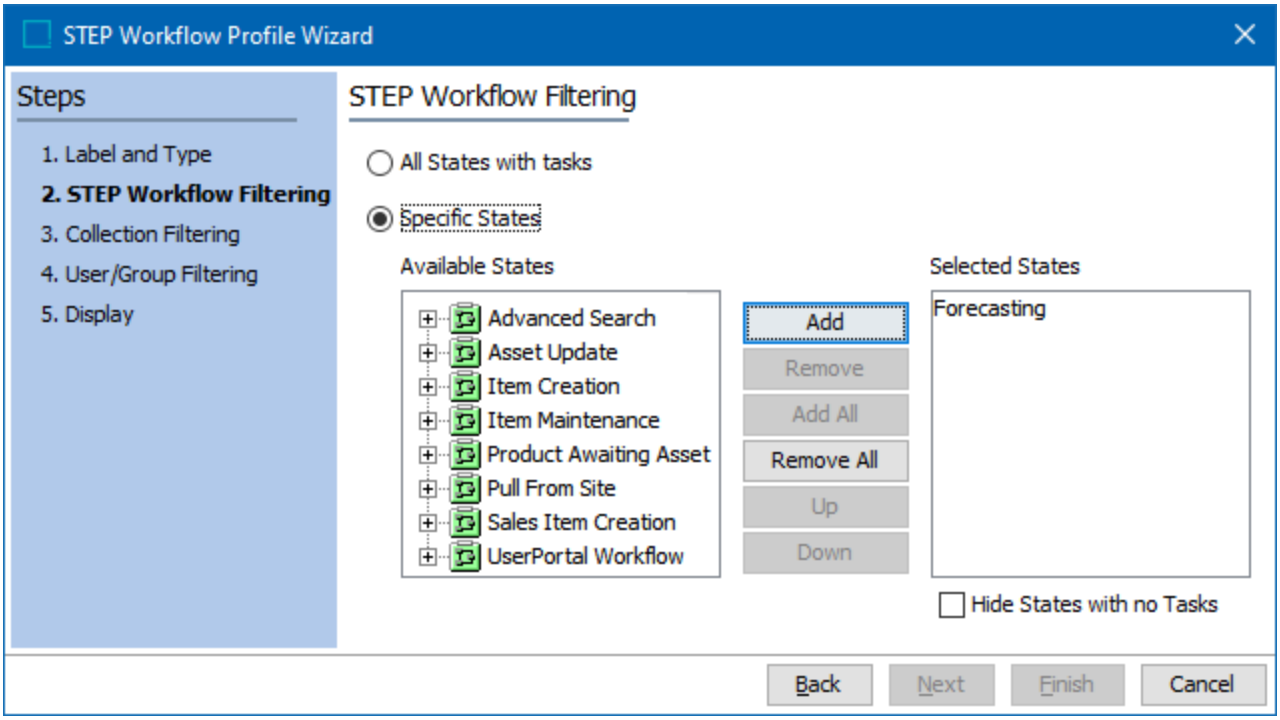
Domain	Description	Example
States	Allows a user to view data focused on workflow states.	• How many objects are in the various states of a workflow • Who the state tasks are assigned to • How long the objects have been in the states • Number of exceeded deadlines • How long the objects have been in the states that are assigned to a particular user or group • How many objects have passed through the states today versus yesterday, versus the day before, etc.
Assignee	Allows a user to view data focused on assignees.	• How many tasks are assigned to the various assignees • How long the tasks have been assigned to the assignees • If the assignees have tasks with exceeded deadlines • How many tasks the assignee has completed this week versus the week before, etc.
STEP Workflows	Allows a user to view data focused on entire workflows.	• How many objects are currently in the workflows • How long the objects have been in the workflow, on average • What is the workflow throughput this month versus the last six months, etc.

2. For STEP Workflow Filtering, determine the workflows or workflow states (within one or multiple workflows) that the profile should contain as defined below.

The options differ somewhat for each domain (selected in the previous step). Since profiles are not limited to a single workflow, a user could have an Assignee-domain Profile that shows information for tasks in numerous workflows.

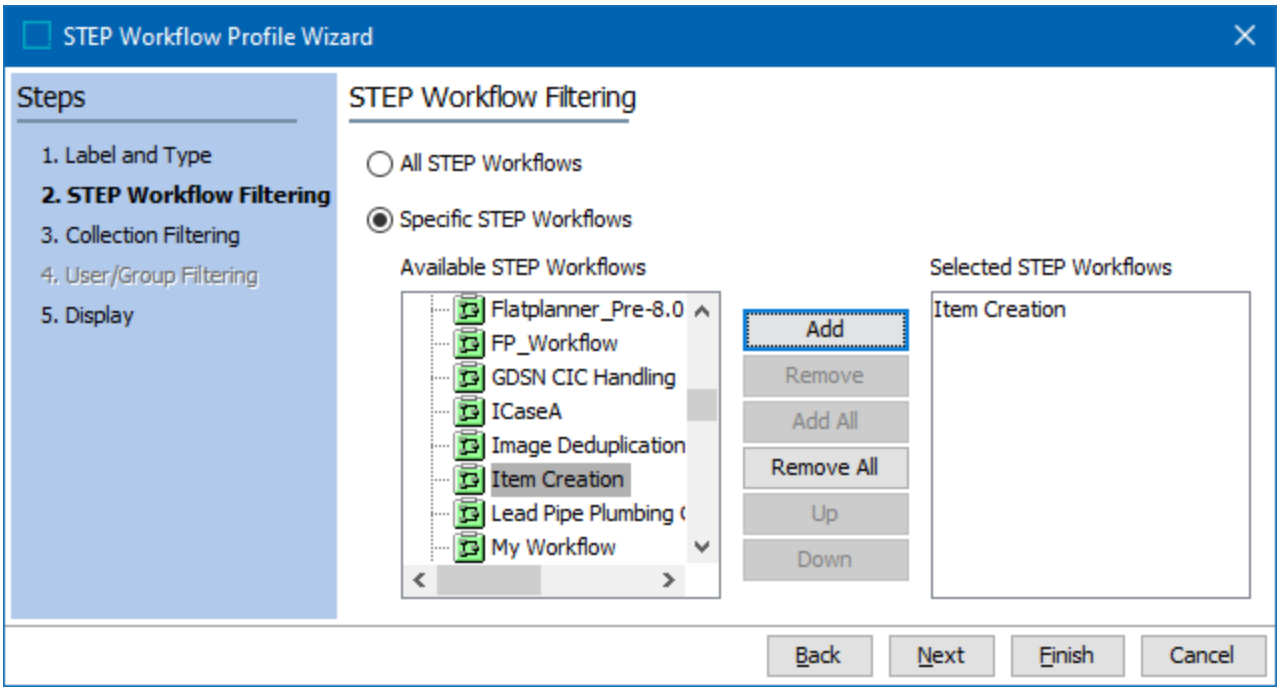
The domain chosen previously impacts the options available in this step.

- **States domain** or **Assignee domain**: Specify whether the profile should contain data from *all* workflow states or if it should be limited to specific workflow states only. If 'All States with tasks' is selected, the profile will contain data from every workflow state, from every workflow in the system. The 'Hide States with no Tasks' checkbox is only available when Specific States are selected.



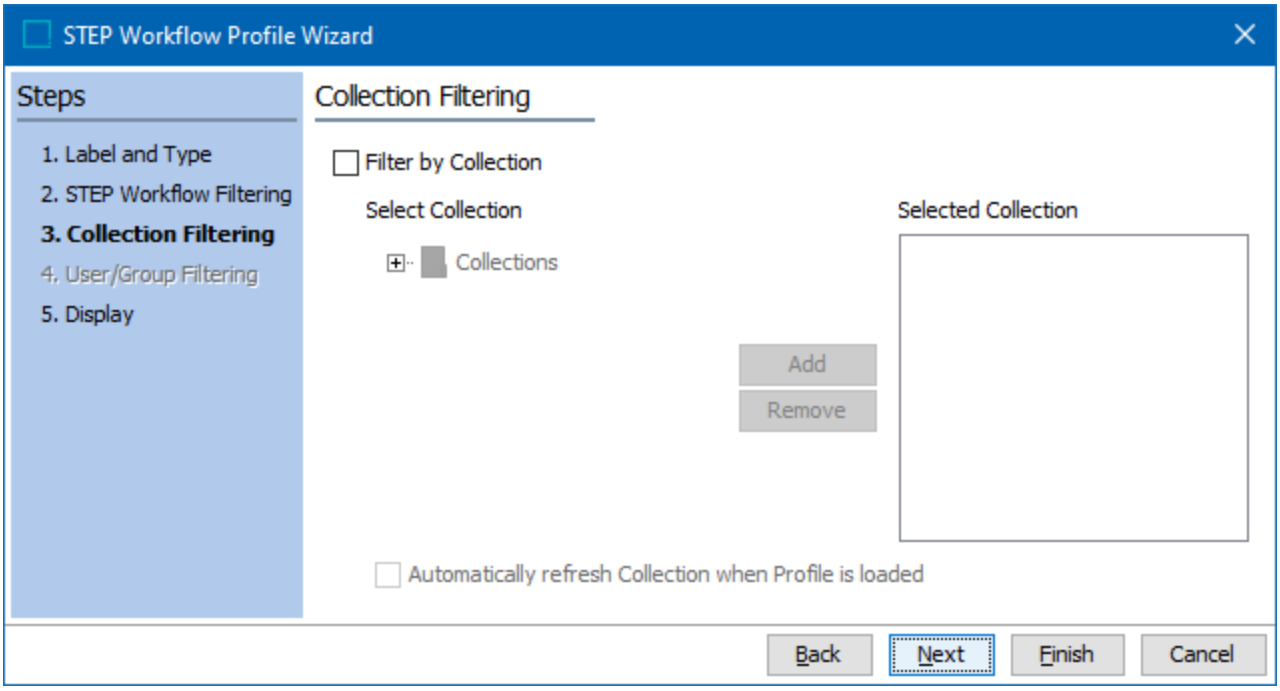
- **STEP Workflows domain**: Only entire workflows can be chosen for this option. Data cannot be filtered

down to the task level.



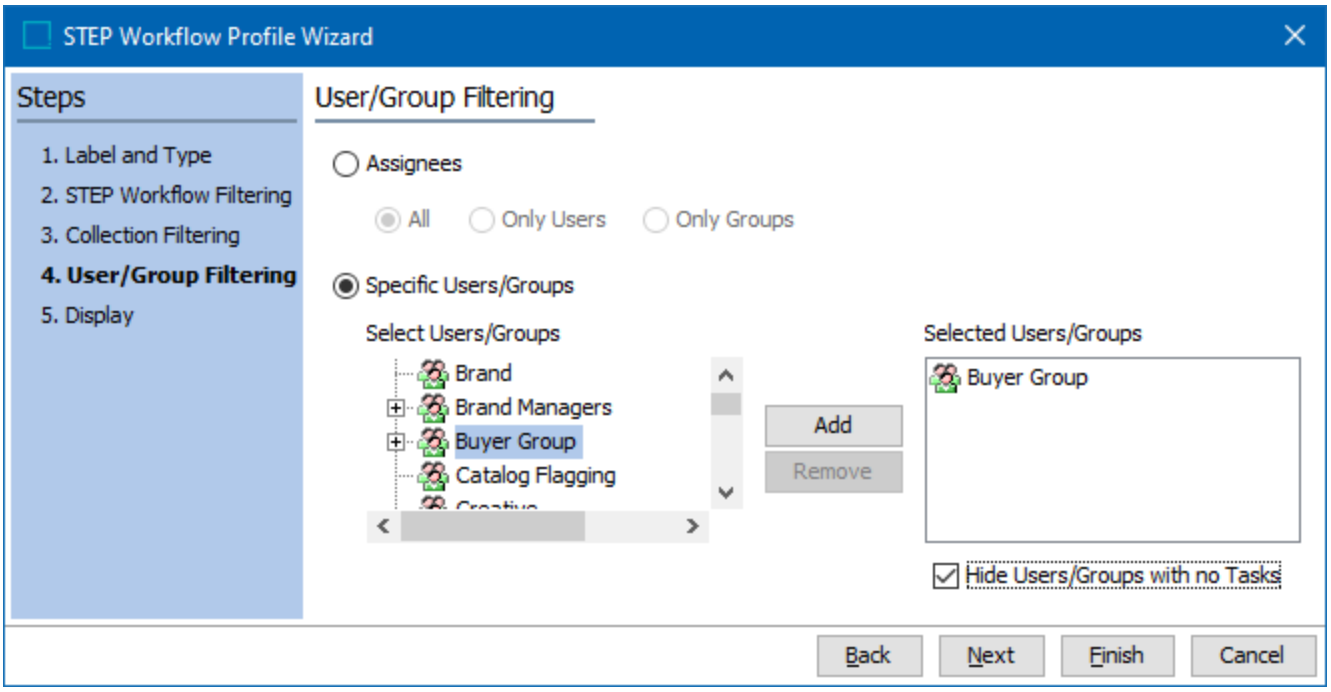
3. For Collection Filtering, a user can optionally filter the profile results to contain information only for objects in a given collection.

   For example, a user could create a profile that only shows data for objects that are to go live at a certain date, or only shows data for objects in a particular part of the tree hierarchy.



4. For User / Group Filtering, a user can filter the profile results to contain information only for tasks that are assigned to specific users or user groups. This step is disabled for the STEP Workflows domain.

   For example, in the **Assignee** domain, the 'Specific Users / Groups' option is valuable to view the workload for a number of users. If the 'Assignees' option is selected, users with *no* tasks will also be included in the profile.



5. For Display, the profile options are different depending on the domain was selected in step 1.

   - For the **States** and **Assignees** domains, the profile will consist of two tables: a **Master Table** and a **Details Table**.

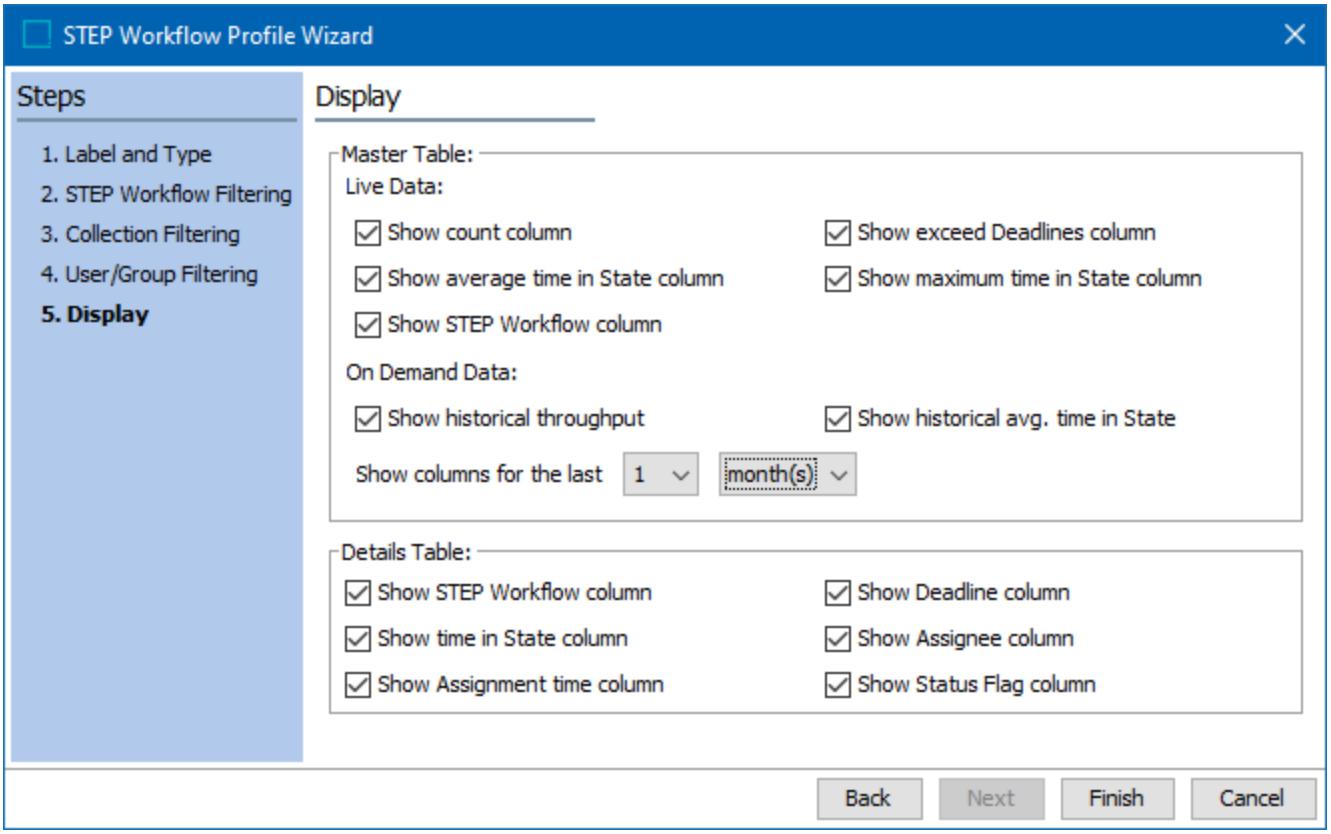
Table	Description
Master Table	Displays more general information about the states or assignees.
Details Table	Appears when a row in the Master Table is selected—displays more specific information about the objects that are contained within a state or assigned to an assignee.

   - For the **STEP Workflows** domain, there is a single Master Table. Step 5 is where a user specifies which columns should be shown in these tables.

Common to all domains are the two divisions of the Master Table settings shown below.

Table Settings	Description
Live Data	Data that can be kept up to date automatically — data for workflow instances in their present state.
On Demand Data	Historical data that is retrieved from the object's State Logs.

- For On Demand Data, this data is generated on demand for all three domains and allows the user to specify a time period of days, weeks, or months for which they would like to generate historical data. For example, if you select three (3) days, the generated On Demand table will display three columns, one for each selected day (e.g., "Throughput today," "Throughput yesterday," "Throughput 2 days ago.")



The following sections explain the different options by domain, for step 5.

States Domain

Table Settings	Description
Master Table-Live Data	• Show count column: Number of objects currently in the state • Show average time in State column: Average time that the objects have been in the state • Show STEP Workflow column: Name of the workflow to which the state belongs • Show exceeded Deadlines column: Number of exceeded deadlines for objects in the state, followed by the average exceed time • Show maximum time in State column: The longest period an object currently in the state has been there
Master Table-On Demand Data	• Show historical throughput: Number of objects that have exited the state within the given period (day/week/month). The number is increased each time an object exits the state, regardless of whether the object has been in and exited the state earlier. • Show historical average time in state: Average time that objects—which have exited the state within the given period (day / week / month)—were in the state
Details Table	• Show STEP Workflow column: Title of the workflow to which the state selected in the Master Table belongs • Show time in State column: How long the object has been in the state • Show Assignment time column: How long the task (which the object-in-state represents) has been assigned to the current assignee • Show Deadline column: The deadline, if any • Show Assignee column: The assignee • Show Status Flag column: The current status

Assignees Domain

Table Settings	Description
Master Table-Live Data	• Show count column: Number of tasks currently assigned to the User/Group • Show average Assignment column: Average time the current tasks have been assigned to the user / group • Show exceeded Deadlines column: Number of exceeded Deadlines for Tasks assigned to the user / group, followed by average exceed time • Show maximum Assignment time column: Longest period that a task has been assigned to the user / group
Master Table-	• Show historical number of assignments: Number of assignments within the given period

Table Settings	Description
On Demand Data	(day / week / month). i.e., how many times during the period have tasks been assigned to the user / group • Show historical number of completed tasks: Number of tasks that were completed within the given period (day / week / month) while assigned to the user / group • Show historical average assignment time: Average time tasks were assigned to the user / group within the given period (day / week / month) • Show historical average completion time: Average time within the given period (day / week / month) that it took a user / group to complete a task, once assigned
Details Table	• Show STEP Workflow column: Name of the workflow in which the task exists • Show State column: The state in which the task exists • Show Assignment time column: How long the task has been assigned to the current assignee • Show Deadline column: The deadline, if any • Show time in State column: How long the task has existed, i.e., how long the object has been in the state • Show Status Flag column: The current status

STEP Workflows Domain

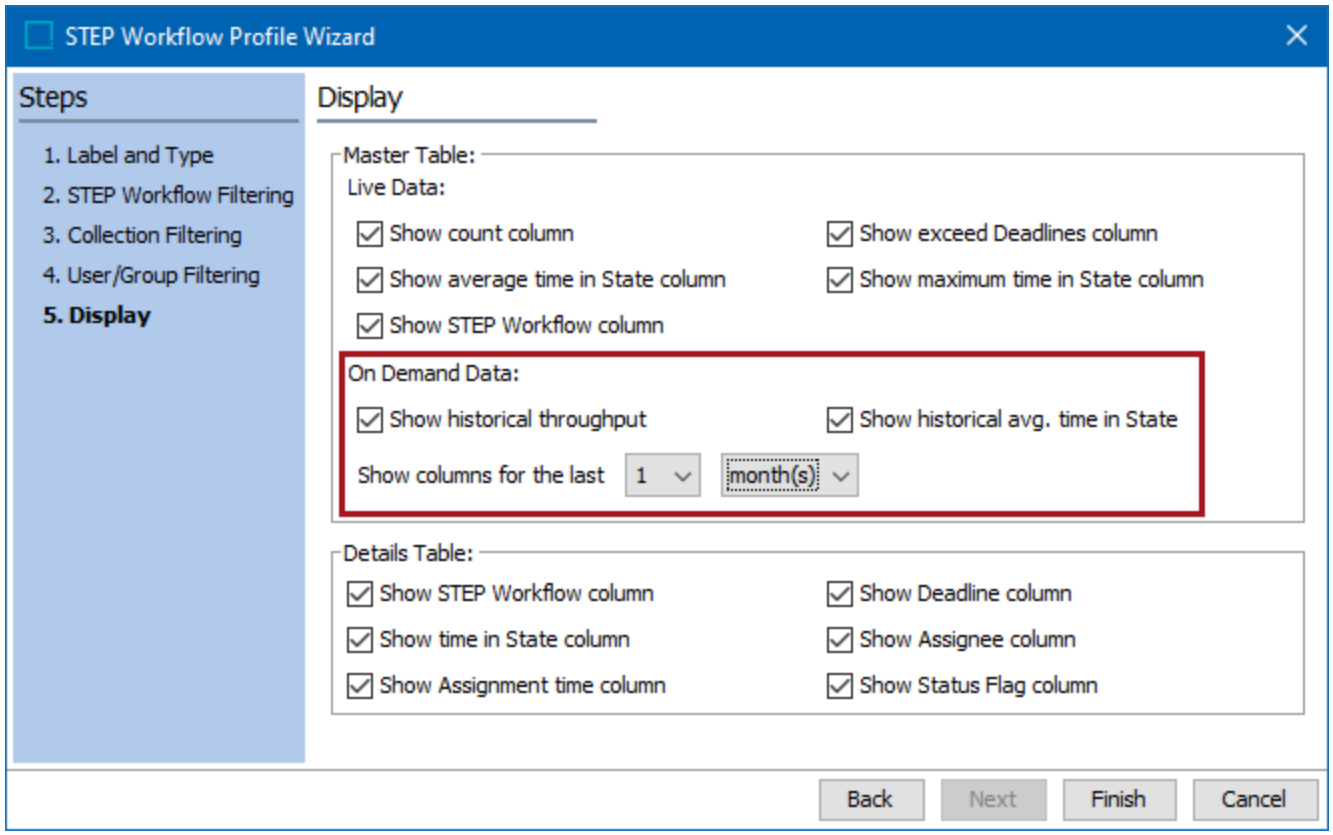
Table Settings	Description
Master Table-Live Data	• Show count column: Number of objects currently in the workflow that are not in a final state on the outer level • Show average time in STEP Workflow column: Average time the objects have been in the workflow
Master Table-On Demand Data	• Show historical throughput: Number of objects that have reached a final state on the outer level during the given period (day / week / month) • Show historical avg. time in STEP Workflow: Average time that it took objects to reach a final state on the outer level (from time of workflow initiation) during the given period (day / week / month)

6. Click the **Finish** button.

Profile Editor

Once a profile has been configured, you can generate the On Demand data by clicking the 'Update on-demand data' button at the top of the Profile tab. You can also edit the configuration and copy the master table data into the clipboard, so you can paste it into a spreadsheet (e.g., Excel).

Note: The 'Update on-demand data' button option is available only if at least one of the 'On Demand Data' checkboxes are checked. If the button is not available, edit the configuration to enable it.



Clicking the 'Update on-demand data' button creates a background process, and updates the table. This displays the number of objects that have exited the workflow (for STEP Workflow domain) within the given period or based on the average time in state.

Note: Data from Workflow Profiles can be represented in widgets on the Global Dashboard in workbench. For more information, see the **Global Dashboard** topic in the **Data Profiling** documentation.

Remove Objects from Workflows

There are many reasons why an object may need to be removed from a workflow. A few of the reasons for removing an object from a workflow are:

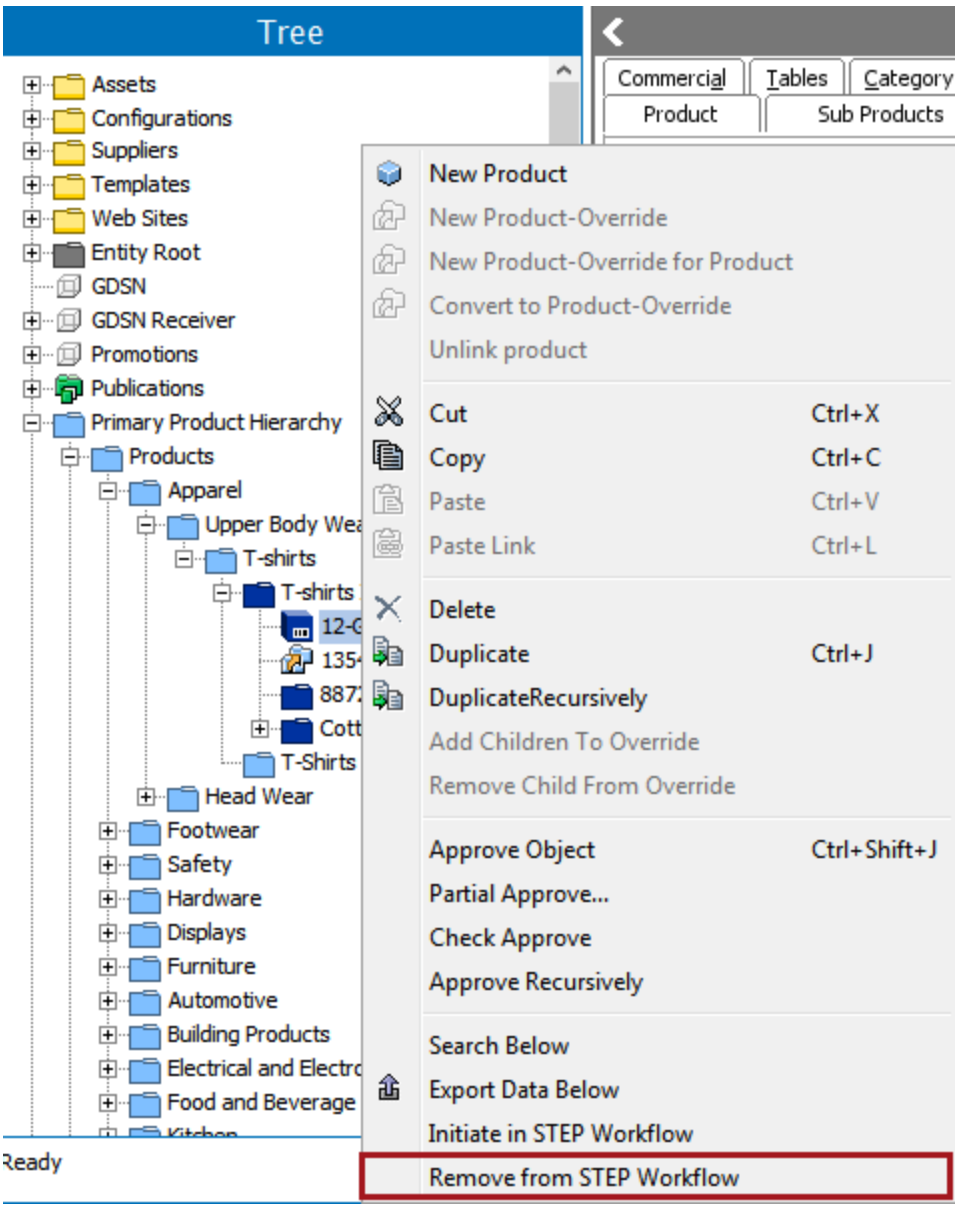
1. It has finished going through the workflow
2. The object needs to be amended before being put back into the workflow
3. The object is no longer relevant for the workflow

No matter the reason for removing an object from a workflow, objects can be removed, i.e., taken out of workflows, in several different ways.

Remove Single Object

If a user wants to remove a single object from a workflow, and they have the 'Remove Items from STEP Workflow' privilege, this can be accomplished via the **Remove from STEP Workflow** context menu option on the object.

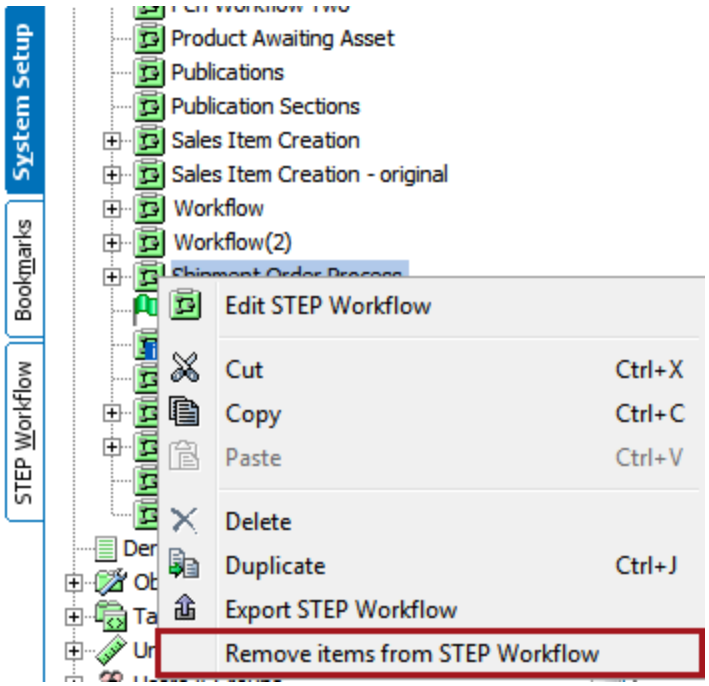
To do so, right-click on an object, or product, and select **Remove from STEP Workflow**.



Remove All Objects from a Workflow

If a user needs to remove all objects from a workflow, and they have the 'Remove Items from STEP Workflow' privilege, this is possible via the workflow context menu option **Remove items from STEP Workflow**.

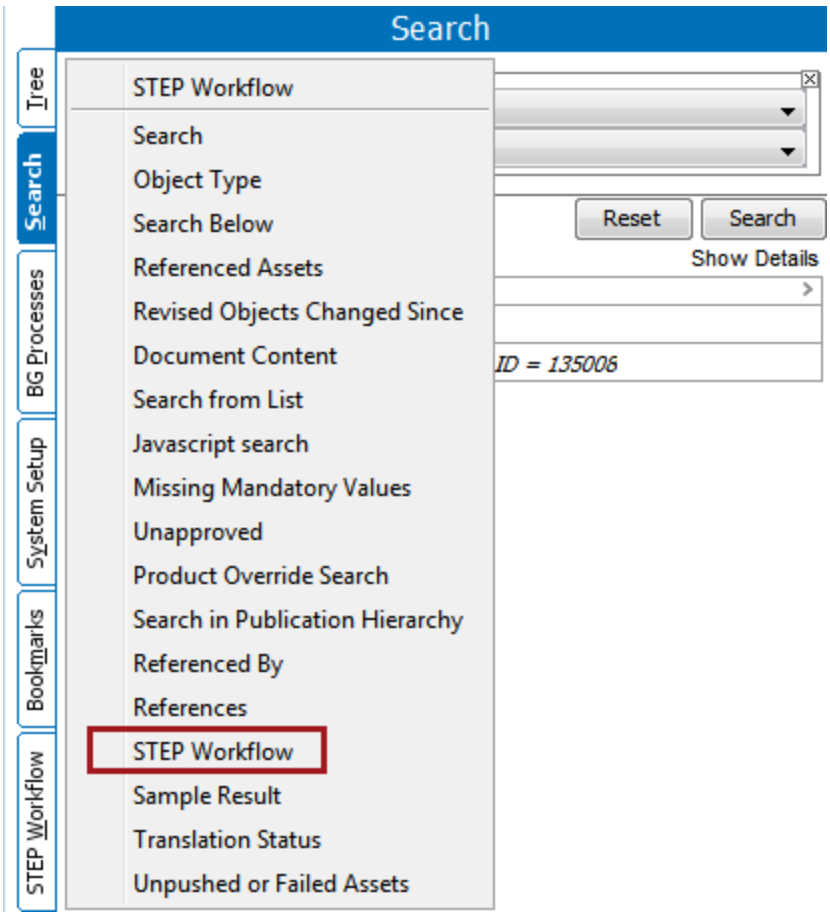
Navigate to System Setup, right-click on the workflow that needs all items removed, and select **Remove items from STEP Workflow**.



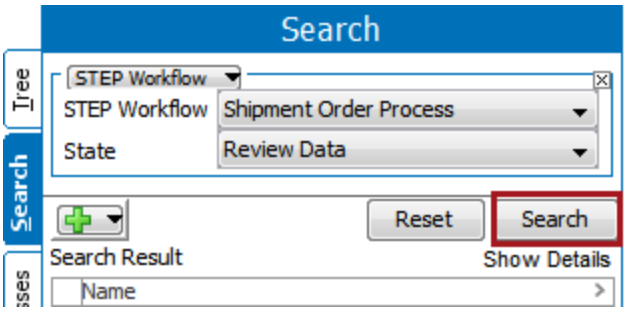
Remove Objects from a State Using Bulk Updates

Should a user want to remove all objects from one state in a workflow, they can do so using Bulk Updates if they have the ability to search the STEP Workflow tab, and they have the 'Perform Bulk Update' privilege.

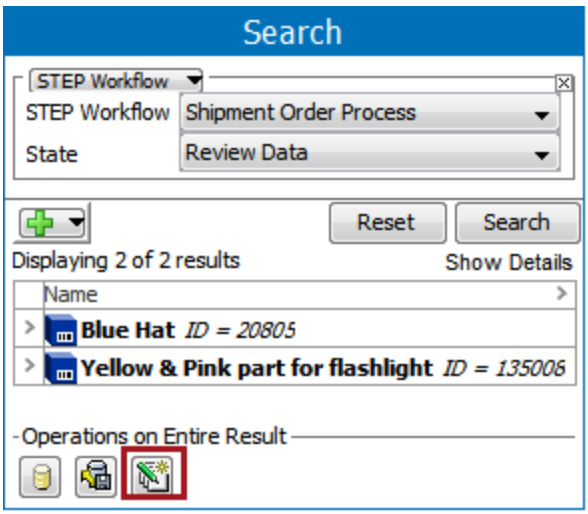
1. First navigate to the main Search tab, and once on the tab, from the dropdown Search menu select **STEP Workflow**.



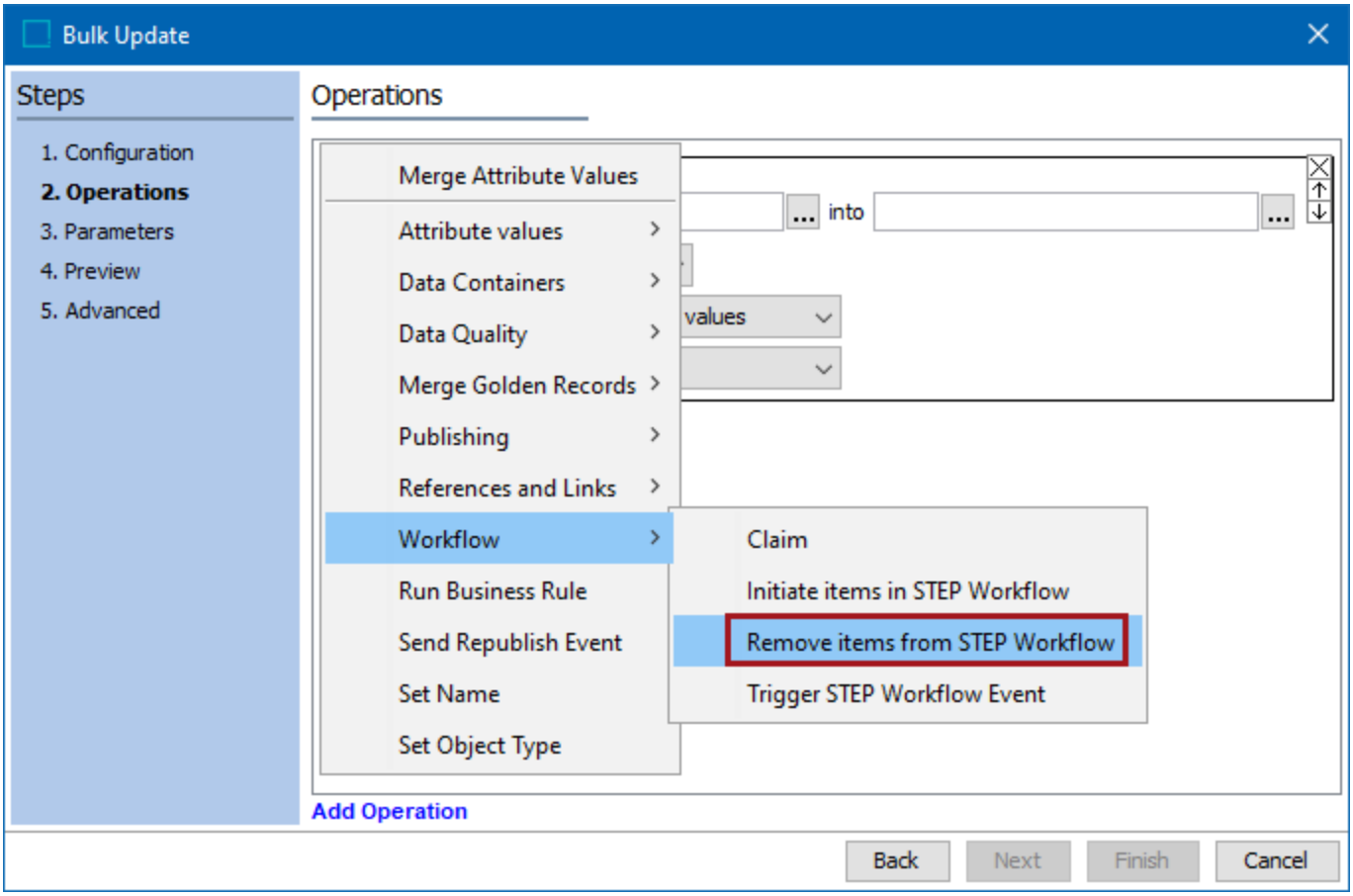
2. Next, from the two dropdown menu, selects the desired workflow and then specifies which state all objects are to be removed from. Press **Search** when finished.



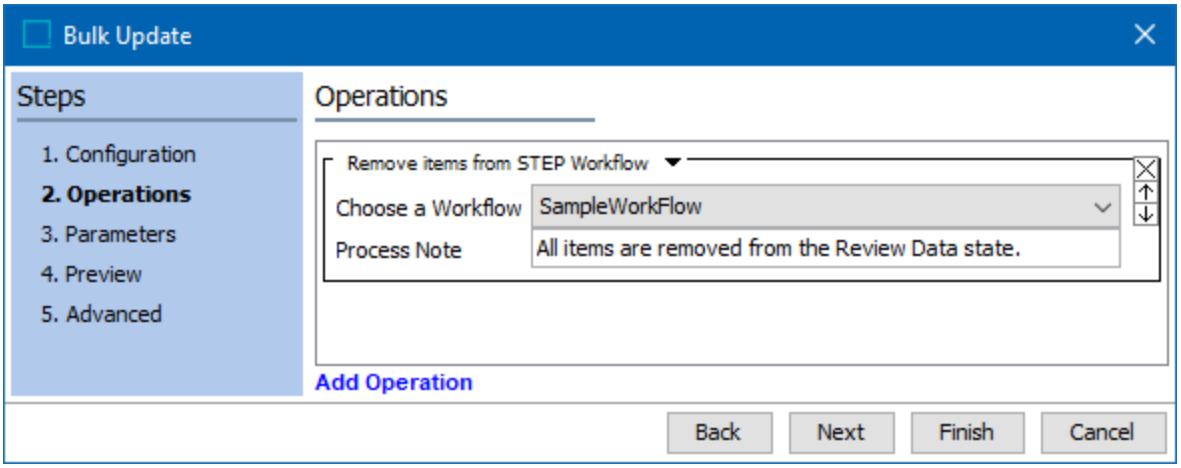
3. Once the search result displays, click on the Bulk Updates button at the bottom of the results list.



4. The Bulk Updates window will display. Click next and on the second step, **Operations**, from the dropdown menu at the top select Workflow > **Remove items from STEP Workflow**.



5. Select the desired workflow, enter a 'Process Note' if desired, and select **Finish**. This will remove all objects from the state.

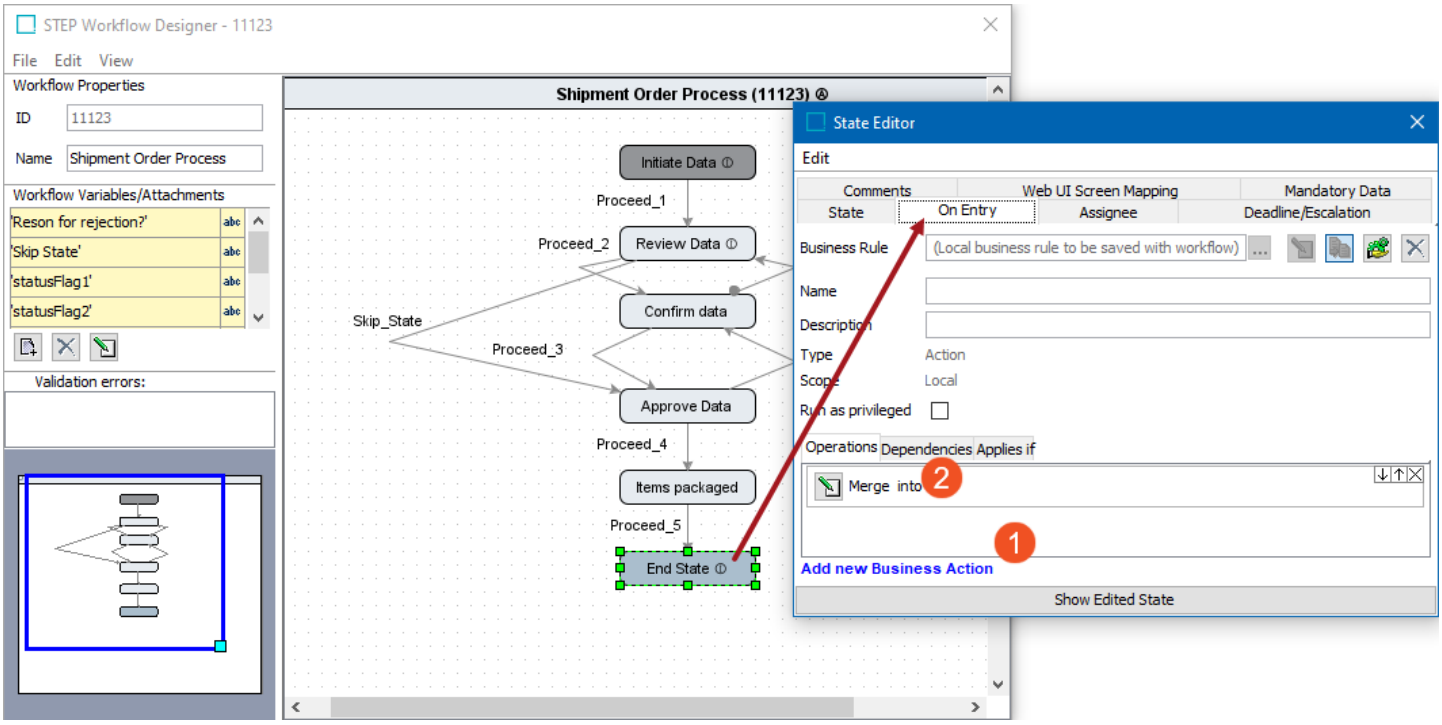


Removing Objects from the Final State

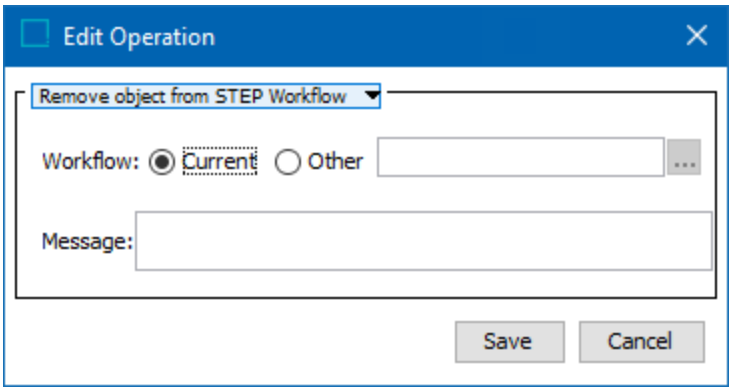
As objects by default stay in the final states of the workflow until explicitly removed, it is highly recommended to automatically take the objects out of the workflow when reaching a final state on the outer level. This can be done one of two ways: use the 'Remove object from STEP Workflow' business action or use an 'Execute JavaScript' business action. It is important to note that both business actions can be used on states other than the final state, however, the information described below is the most typical scenario in which these actions are used.

Remove object from STEP Workflow Business Action

1. To set up the business action, navigate to System Setup, open the STEP Workflow Designer for the desired workflow, and right-click on the outer level **Final State**.
2. In the dialog that appears, select the On Entry tab > Add new Business Action > **Merge into**.



3. Go to 'Workflow' from the dropdown menu, and click on the **Remove object from STEP Workflow** option in the submenu.
4. The **Current** radio button is selected by default. (If using this business action in a way that it will be invoked outside of a workflow or if the current action should apply to a different workflow, then you can select Other and specify a different workflow.)
5. Next, if desired, add a message to be logged in the Note field of the object State Log.



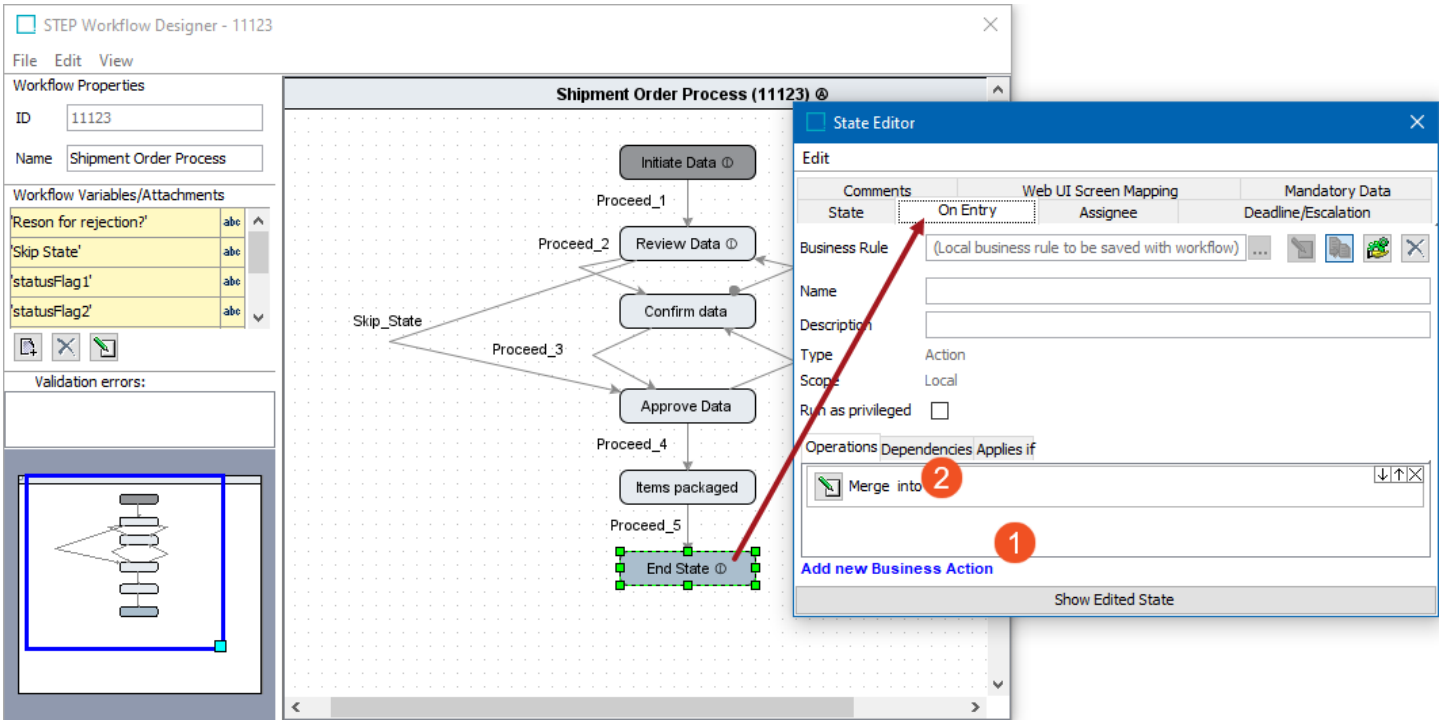
6. Click **Save** and remember to save your changes before exiting the STEP Workflow Designer.

Execute JavaScript Business Action

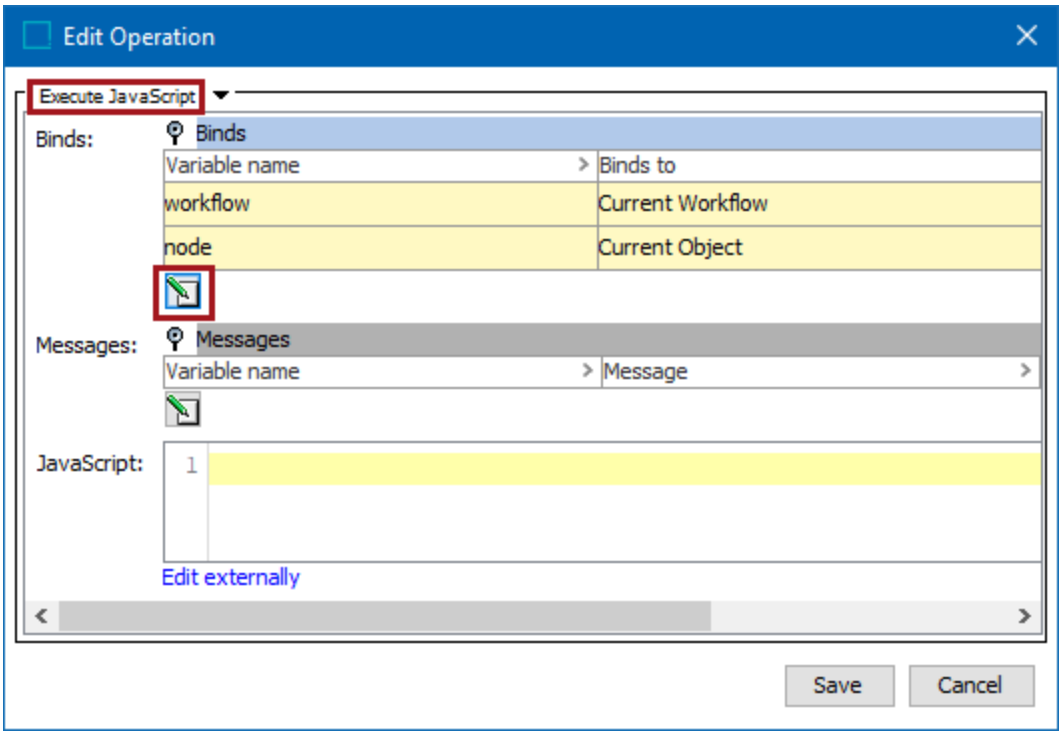
If below script is entered in a business action configured 'On Entry' to the final state, once objects enter the state, they will be removed from the workflow and the supplied process note posted to the State Log if one was entered.

```
node.getWorkflowInstance(workflow).delete("Object automatically removed
from Workflow");
```

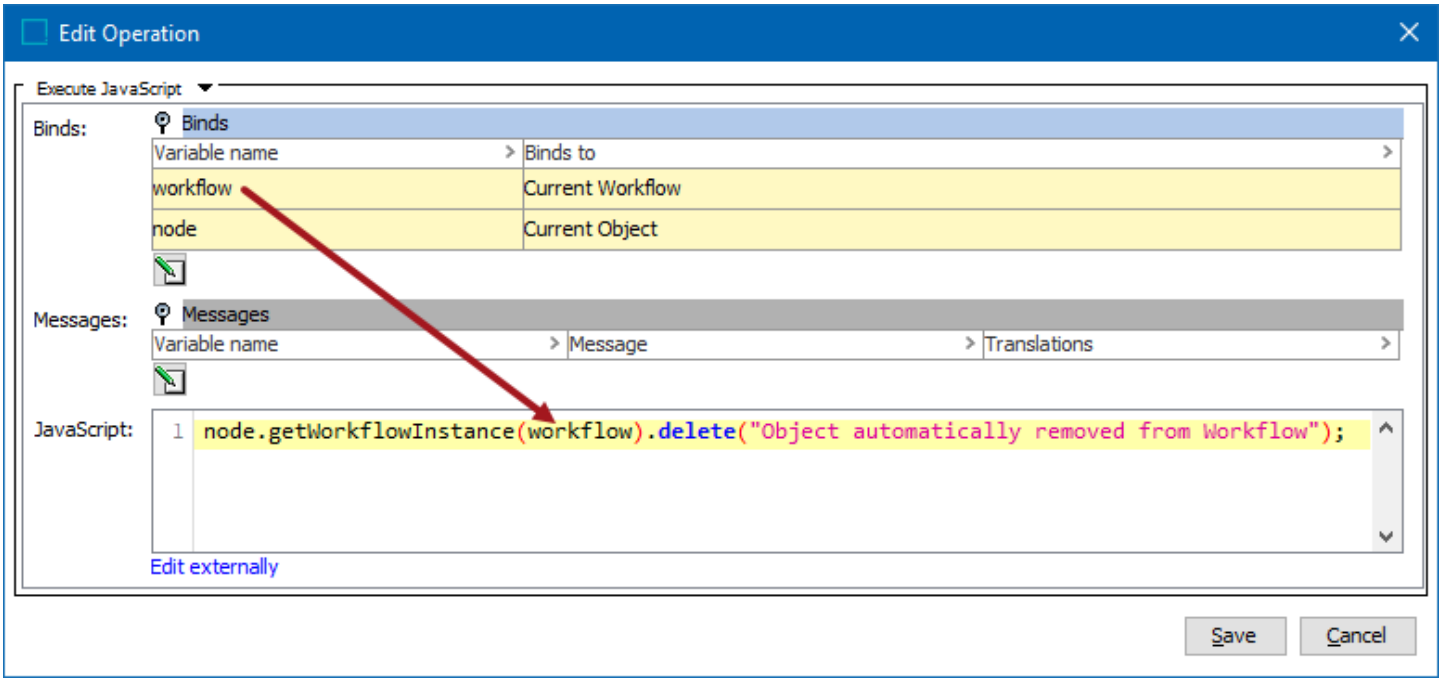
1. To add the above script to a business action, navigate to System Setup, open the STEP Workflow Designer for the desired workflow, and right-click on the outer level **Final State**.
2. In the dialog that appears, select the On Entry tab > Add new Business Action > **Merge into**.



3. Select 'Execute JavaScript' from the dropdown menu, and click on the **Edit Binds** button, to add any necessary binds. Note that the two binds being added for this particular workflow are: Workflow to Current Workflow and Node to Current Object.



4. Next add the following script in the JavaScript window. Notice how the variable name for the Current Workflow Bind is used. Users should make sure to replace with their variable name in the script.

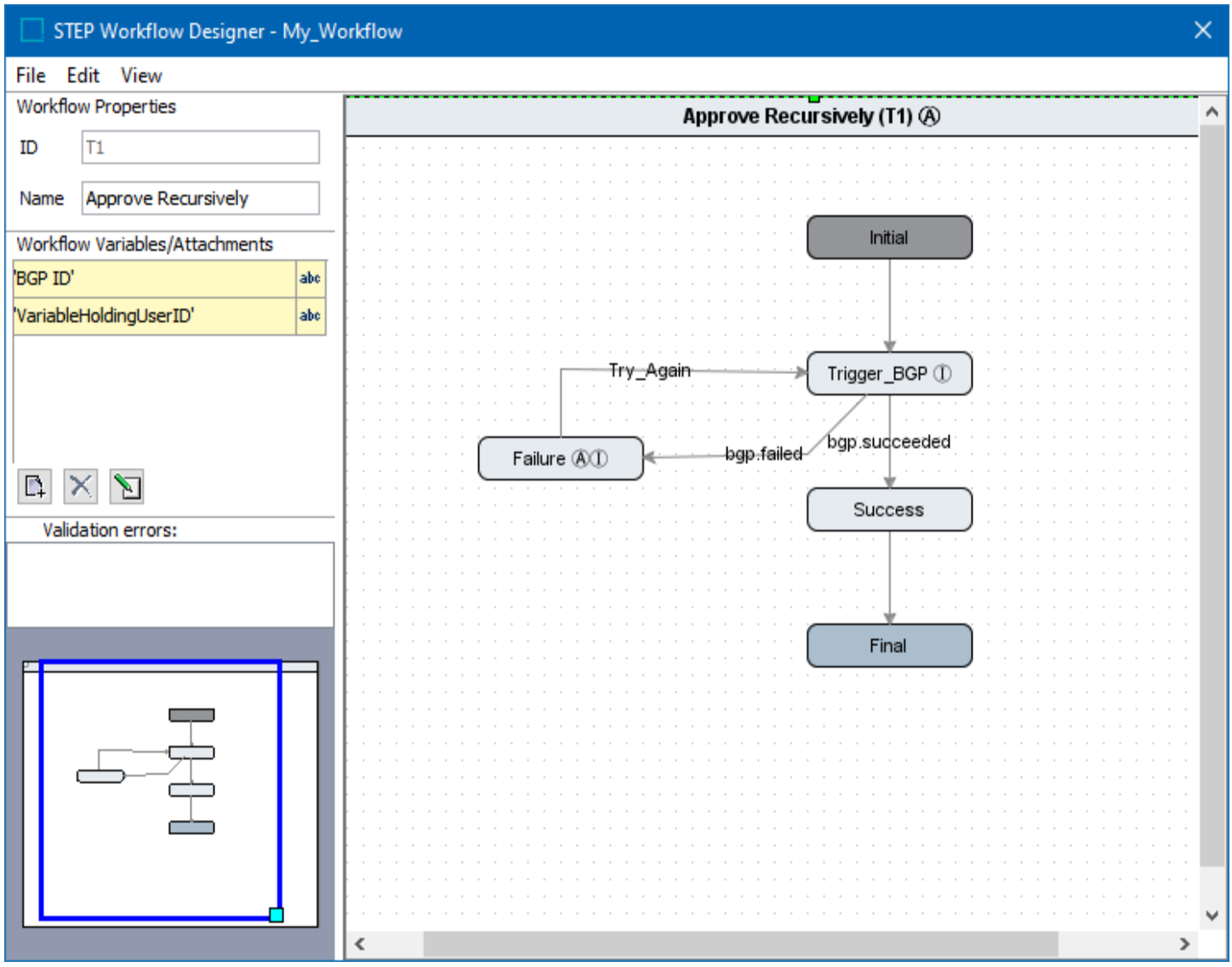


5. Click **Save** and remember to save your changes before exiting the STEP Workflow Designer.

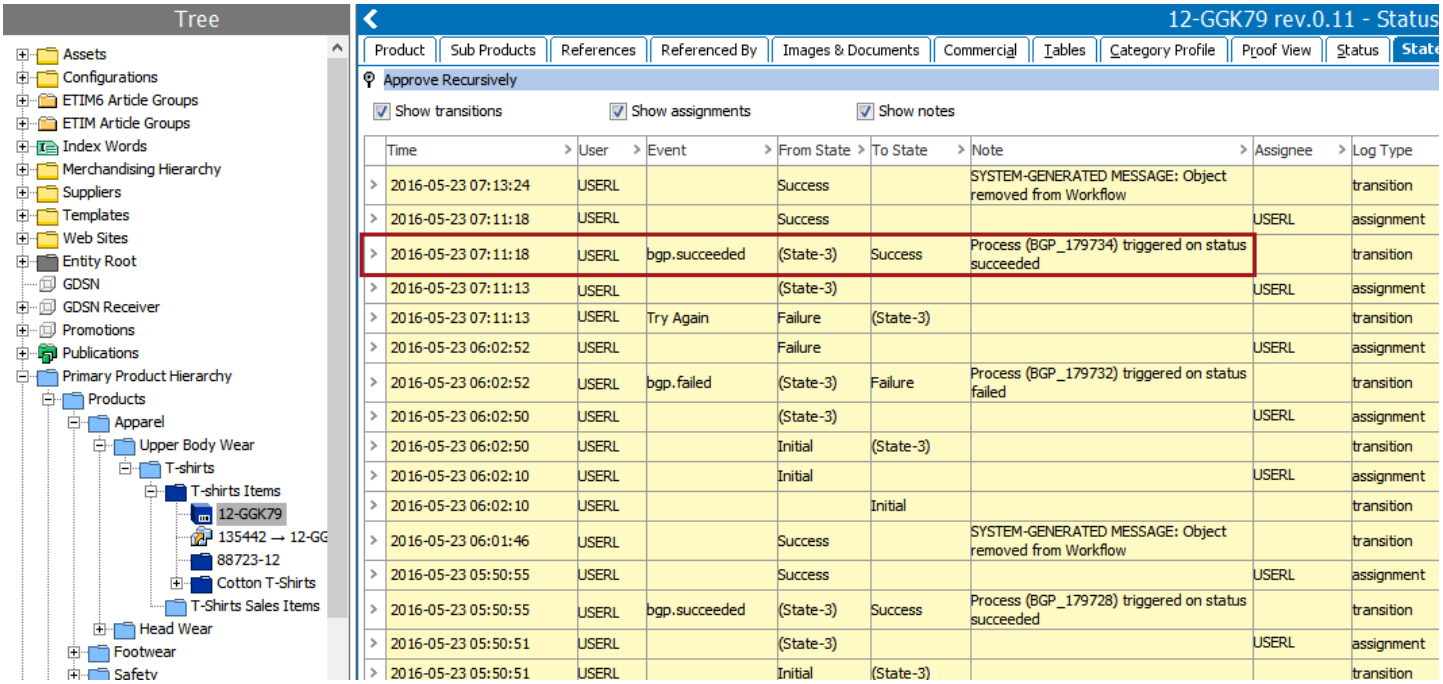
Running Background Processes from Workflows

Via the API Task interface, it is possible to start a background process, BGP, and have events triggered on the task matching the end status of the background process. Image below shows a simple workflow where an Approve Recursively Background Process is started when entering the 'Trigger_BGP' state. When the background process is done, one of the transitions out of the state will be performed based on the end status of the background process. i.e., If the end status is 'succeeded', the "bgp.succeeded" event will be triggered. If it is failed, the "bgp.failed" event will be triggered. Similar to the "StateID.Something" (where 'Something' stands for the user's word choice) and "stibo.import" events, any event beginning with "bgp." will be hidden in the STEP Workflow Items editor and on the Object Tasks tab.

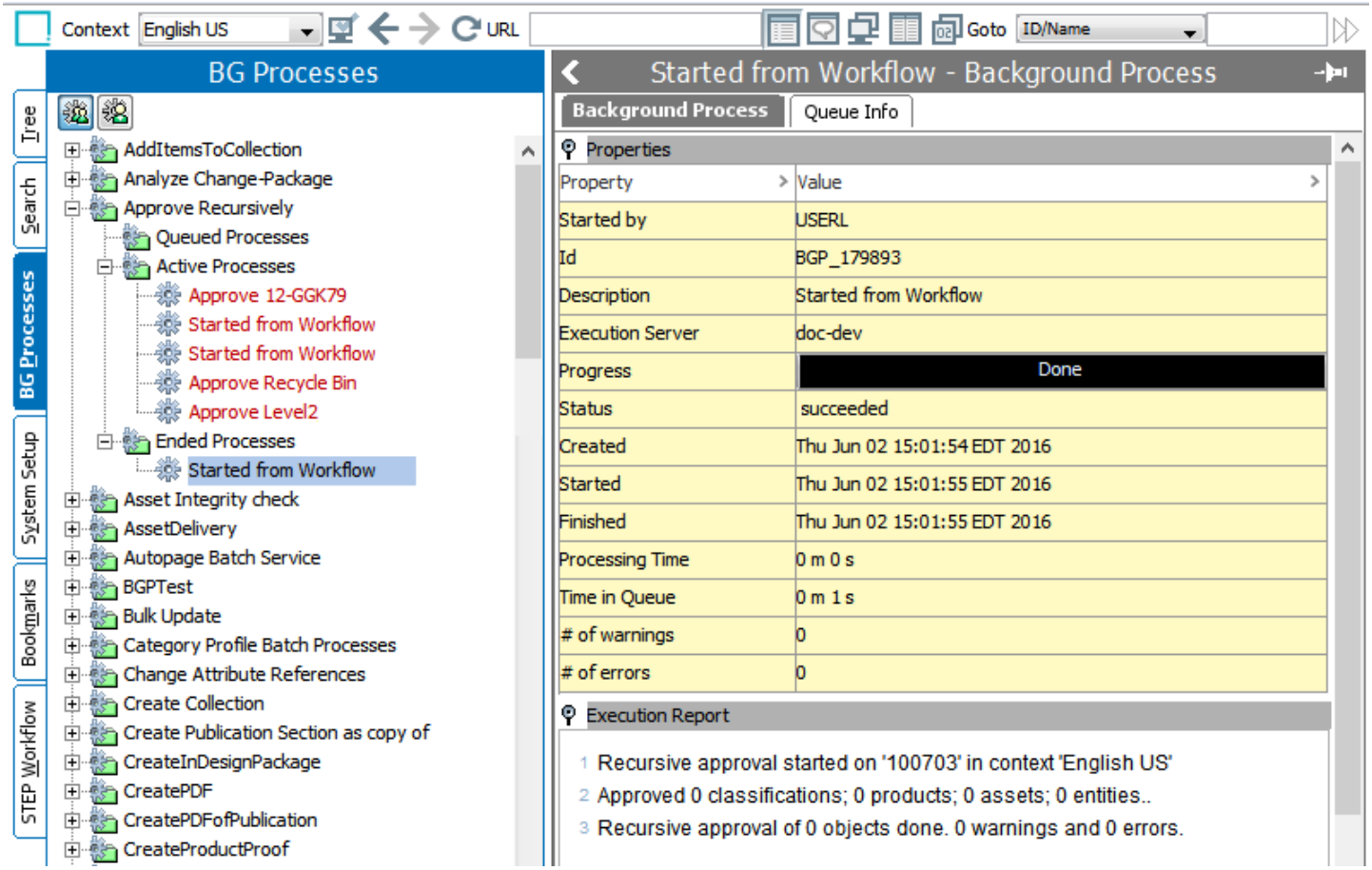
In the picture below, as soon as a task is released to the 'Trigger_BGP' state, the Approve Recursively with a BGP process script will execute.



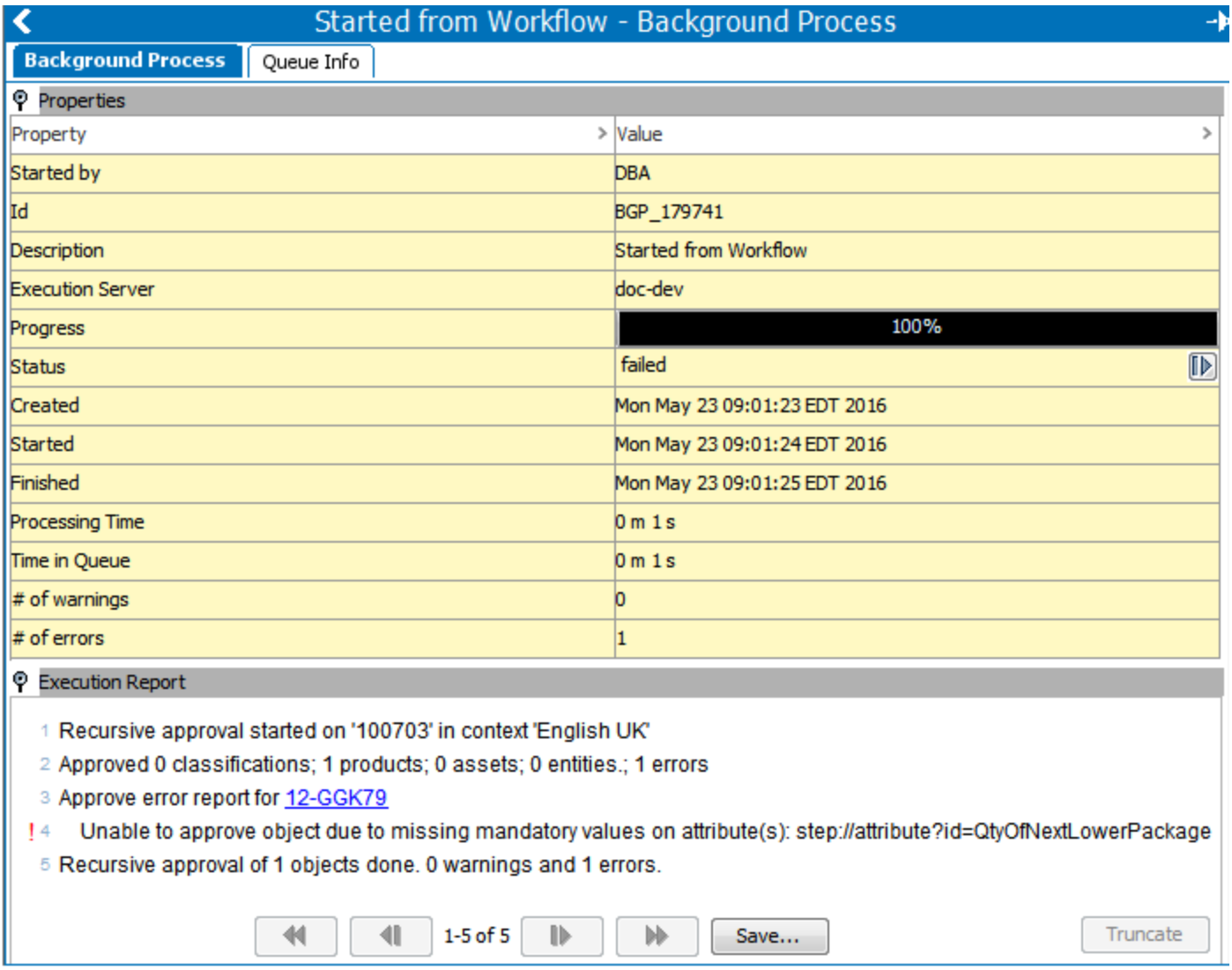
Once a BGP has finished, if a particular task / item needs to be investigated, a user can navigate to its State Log tab and see all Background Process IDs that occurred.



Should the user need to examine any background process information further, they can do so by either going to the main BG Processes tab and looking at the background processes for the receptive workflow or by searching for the Background Process ID.



If the BGP should fail, looking at the BGP report can help in determining why.



In the case of this 'Approve Recursively' workflow example, the task will be moved to the Failure state, and only once the error is fixed, then the task can be moved from the Failure state by selecting the event 'Try_Again'. Then if the BGP is successful, it will move to the Success state.

On Entry Background Process Script

While the script below could be used for other scenarios, two scenarios that this script can be used with workflow for are the following:

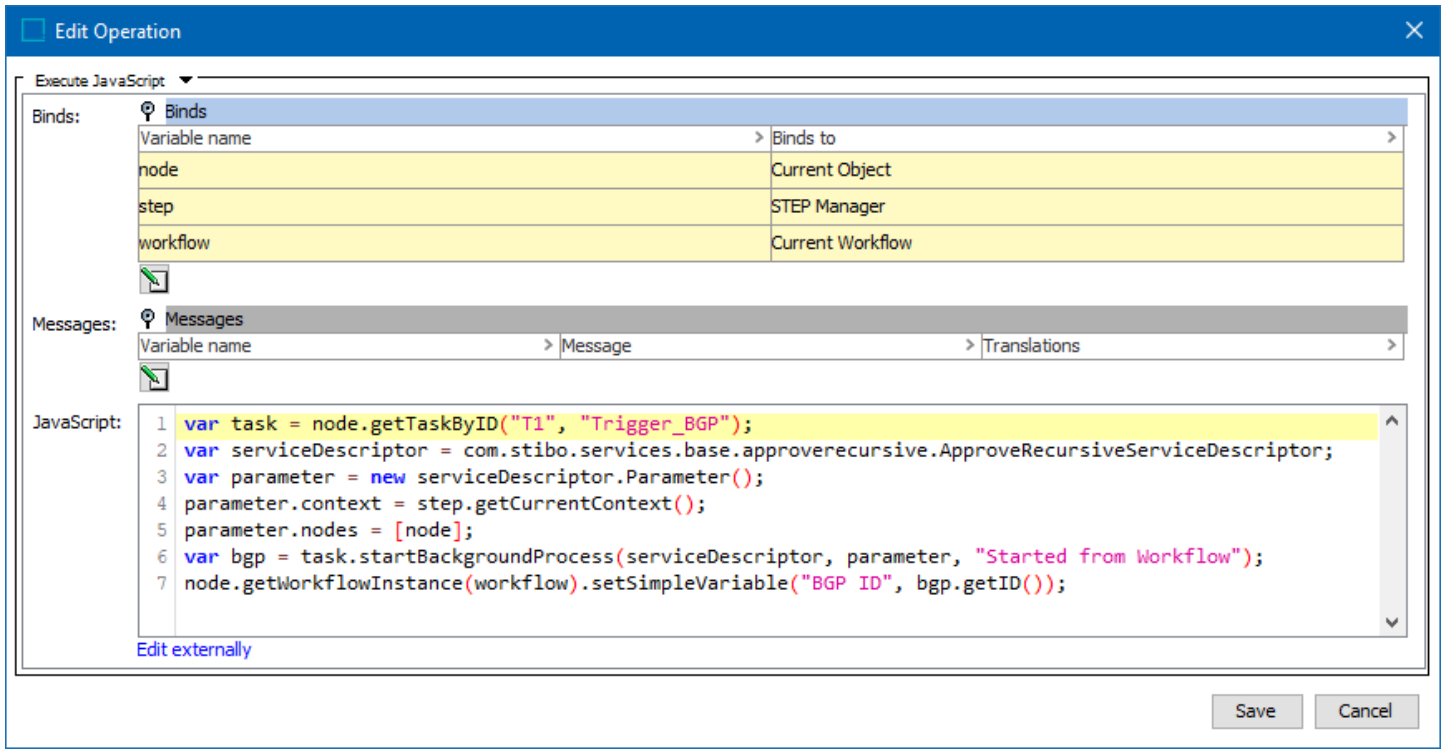
1. If an object is required to run a translation, and it is mandatory that an object is in the approved state before running the translation.
2. If there is any extraction required from the Approved workspace

The On Entry script that starts the background process is shown below. The last line stores the ID of the started background process in a workflow variable with ID "BGP ID". The ApproveRecursiveServiceDescriptor is described in the public API documentation, accessed by clicking the **STEP API Documentation** button on the STEP Start Page.

```
var task = node.getTaskByID("WokflowID", "WorkflowStateID");

var serviceDescriptor =
com.stibo.services.base.approverecursive.ApproveRecursiveServiceDescripto
r;

var parameter = new serviceDescriptor.Parameter();

parameter.context = step.getCurrentContext();

parameter.nodes = [node];

var bgp = task.startBackgroundProcess(serviceDescriptor, parameter,
"Started from Workflow");

node.getWorkflowInstance(workflow).setSimpleVariable
("WorkflowVariableID", bgp.getID());
```

Note: The "WorkflowID" and "WorkflowStateID" in the top line, and "WorkflowVariableID" in the bottom should be replaced by the users desired workflow information.

The On Entry script in the state "Trigger_BGP" will look like the image below. Also notice that the image below has filled in the appropriate "WorkflowID," "WorkflowStateID," and "WorkflowVariableID" in accordance to the workflow information.



Send Email from a Workflow

Generally emails are sent from a workflow to a user to escalate the importance of a task as it moves during the transition from one state to another. There are a number of reasons why a workflow might be setup to do this. One reason might be that a task was rejected from a state by the 'reviewer' due to the task not being done correctly. In this case, if there was a business action set up on the transition to notify the previous user, the user then would receive an email saying that the task they had previously pushed through was sent back. No matter the reason for setting up email notifications from workflows, it is possible to set this configuration up via a business action utilizing the "mailer" script shorthand. The business action can be configured or used on the State Editor screen in the 'Deadline/Escalation', 'On Entry', or 'On Exit' tabs, as well as on the Transition Editor on the 'On Transition' tab.

For information on configuring email from STEP, see the **Email from STEP** topic in the **Resource Materials** of online help.

Dynamic Email Addresses

When sending emails from a workflow, it is often necessary to send emails to different users who interact with the same workflow. One scenario for this might be that a number of users may claim tasks from the same state. If the task is later released, and then rejected due to an error, it is important to inform the correct user who last worked on it.

Configuring Emails from Workflows

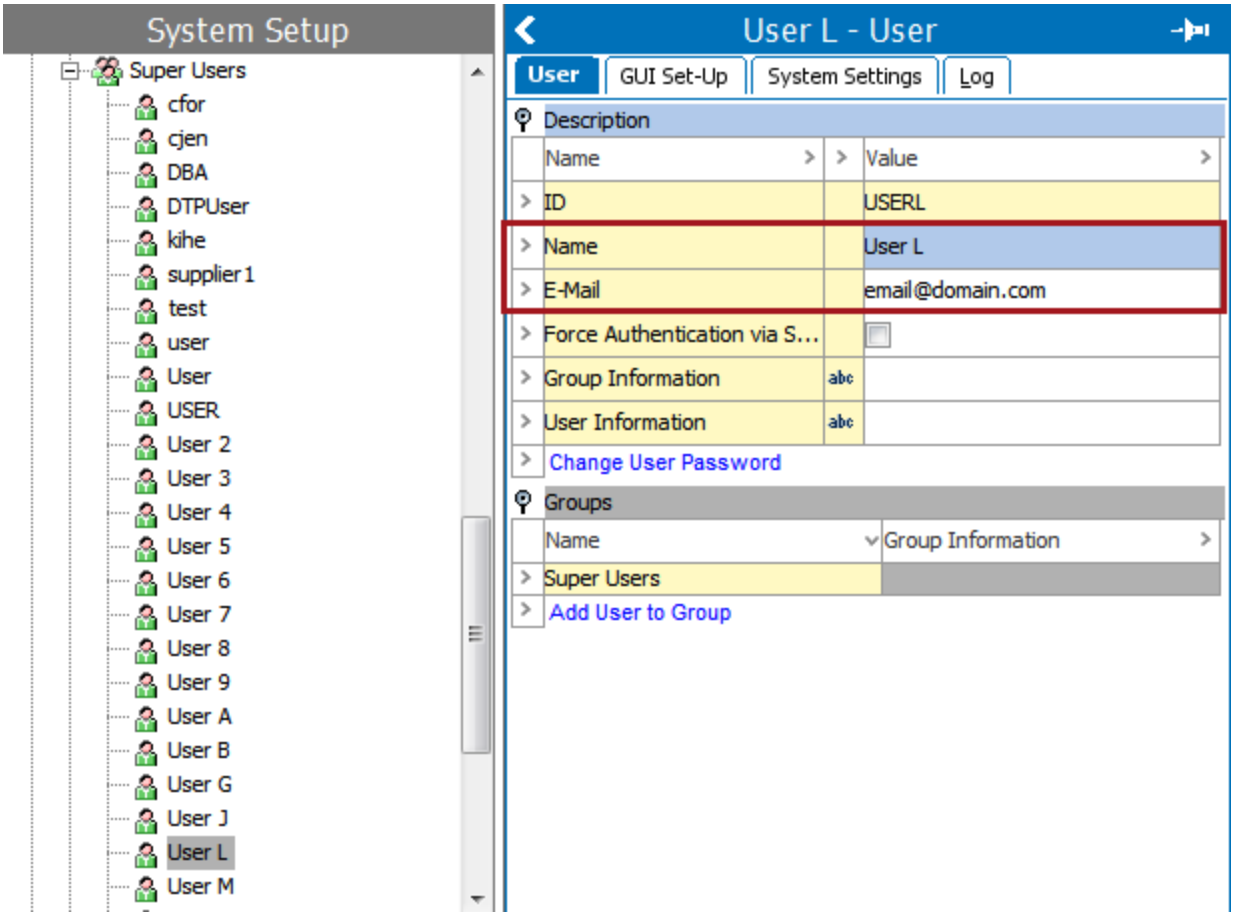
The script example below shows how an email address defined on a STEP User can be obtained, and how an email without attachments is sent to that user. In the following example, this script is used on a Transition Editor on the 'On Transition' tab.

```
var userId = node.getWorkflowInstance(workflow).getSimpleVariable
("VariableHoldingUserID");

var emailAddress = step.getUserHome().getUserById(userId).getEMail();

if(emailAddress) {

        mailer.send("noreply@stibo.com", emailAddress, "From Workflow",

        "This is a mail sent from a Workflow");

}

else {

        logger.warning("No email address specified for User with ID " + userId);

}
```

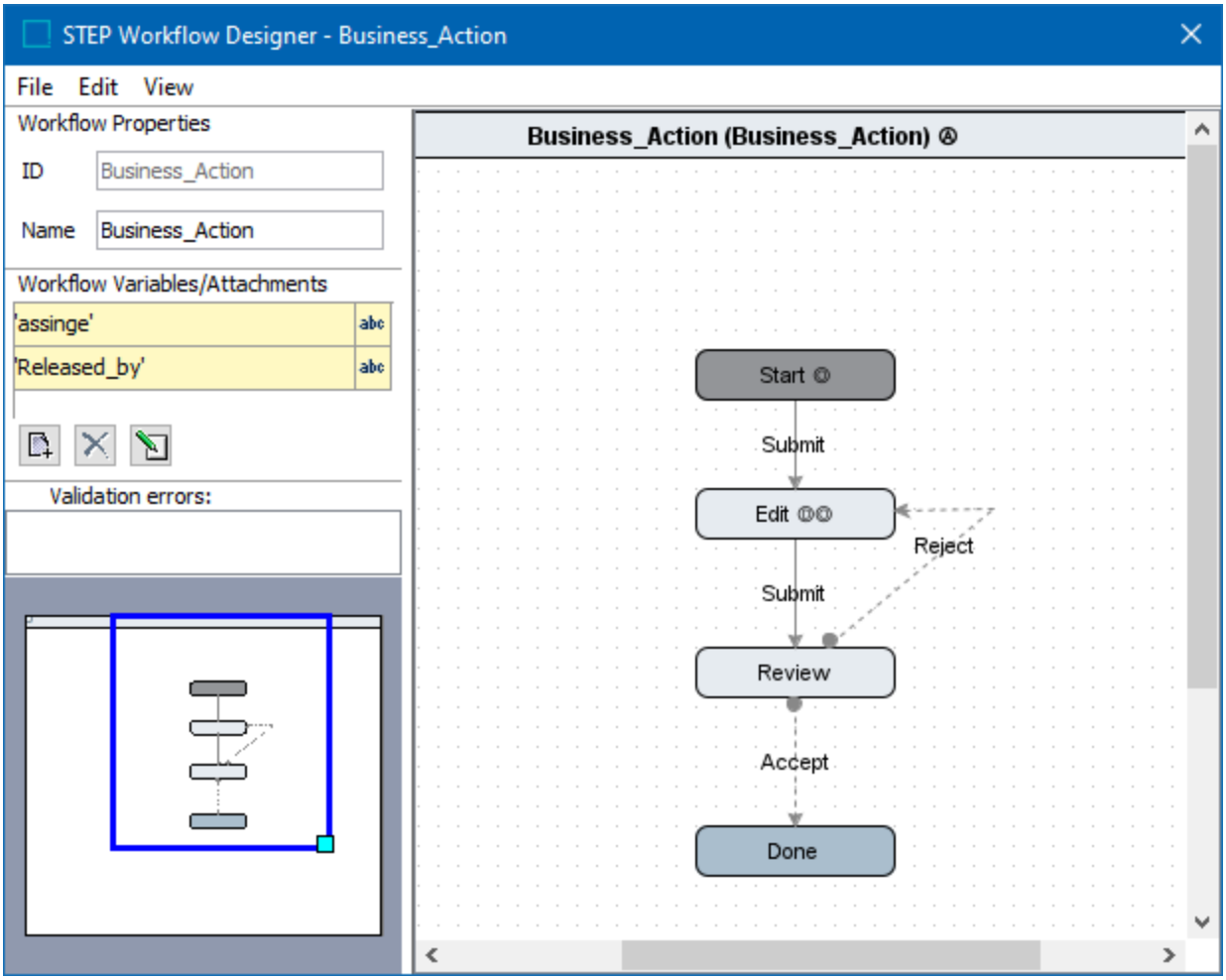
Conditions

To execute the above configuration, the following two conditions must be met for 'Sending Email from a Workflow'.

1. A Simple Mail Transfer Protocol (SMTP) server must be configured in the executing system.
2. The user who is to receive the email should be entered into the User tab under description. To do so, go to System Setup, navigate to the node that houses Users and Groups, expand the node and click on the user who will be emailed. Enter in their name and email.

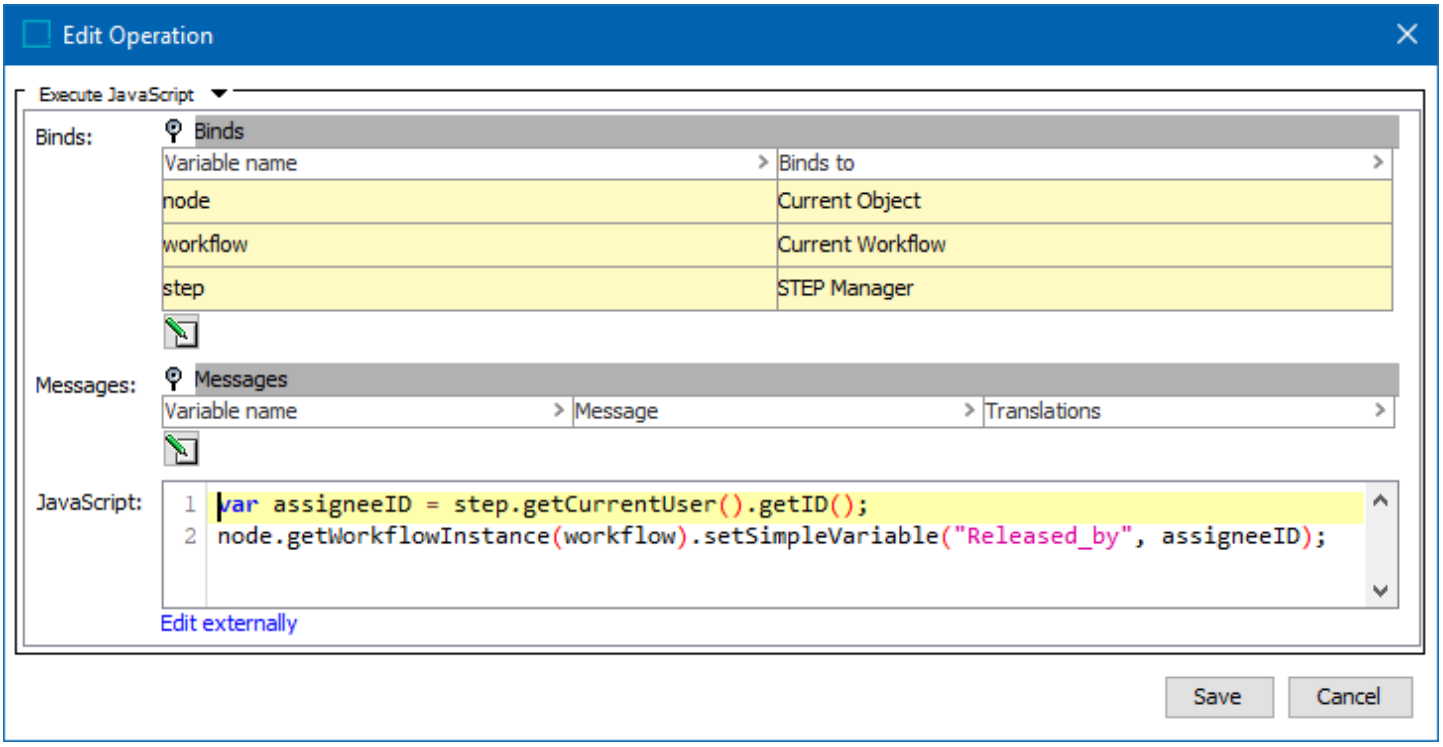


In the example below, there are two business actions set up to send emails to various users.



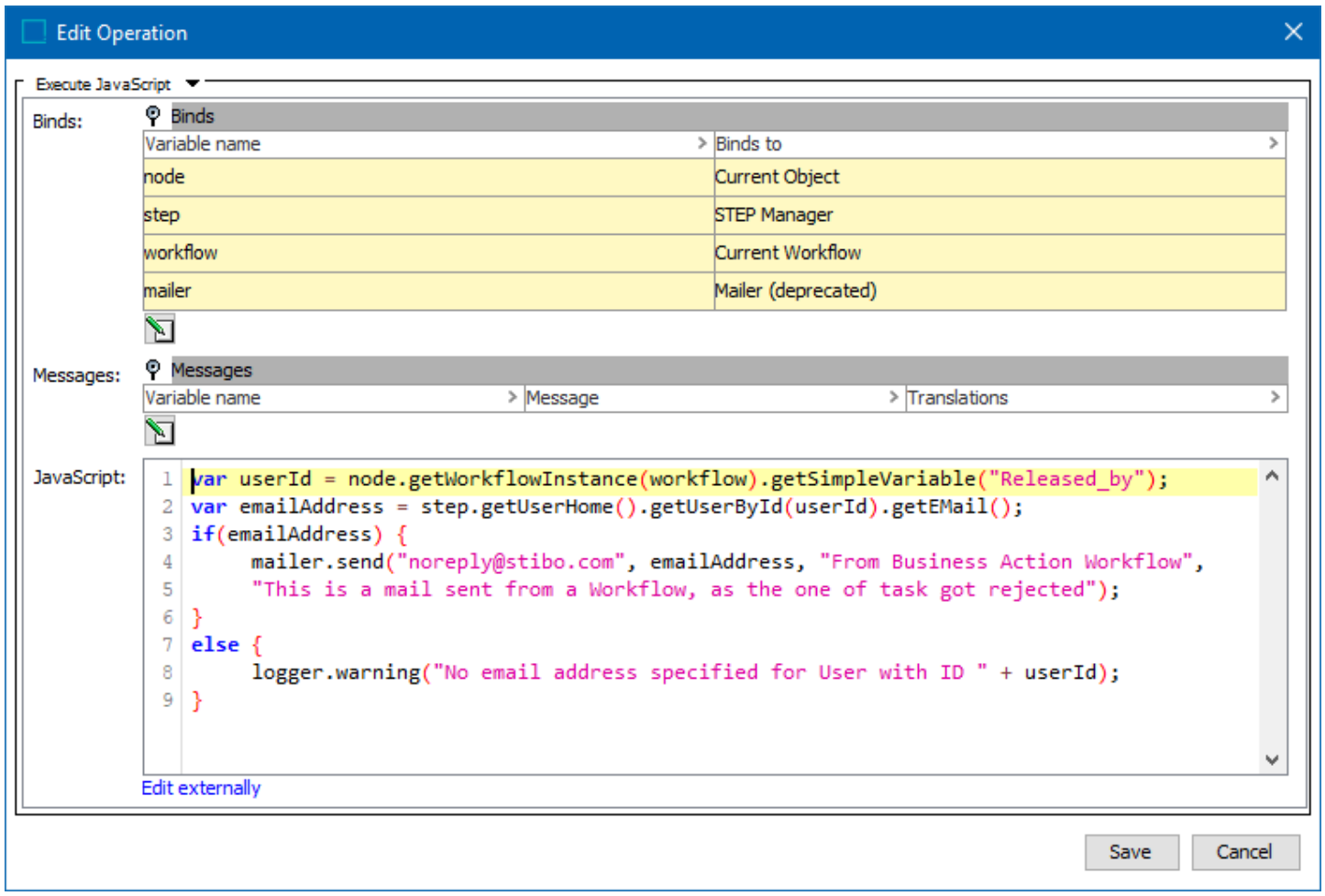
One business rule is on the State Editor on the 'On Exit' tab coming from the Edit state going to the Review state. This business action, which is pictured below, is set up to retrieve the email ID of a user who submits a task from Edit state to the Review state as a variable. In this case, 'Released_by' is the variable defined to retrieve the email ID.

See **Workflow Variables** in the **Workflows** documentation for more on workflow variables.



The second business action is on the Transition Editor on the 'On Transition' tab. This transition is coming from the Review state going to the Edit state and has the event name Reject.

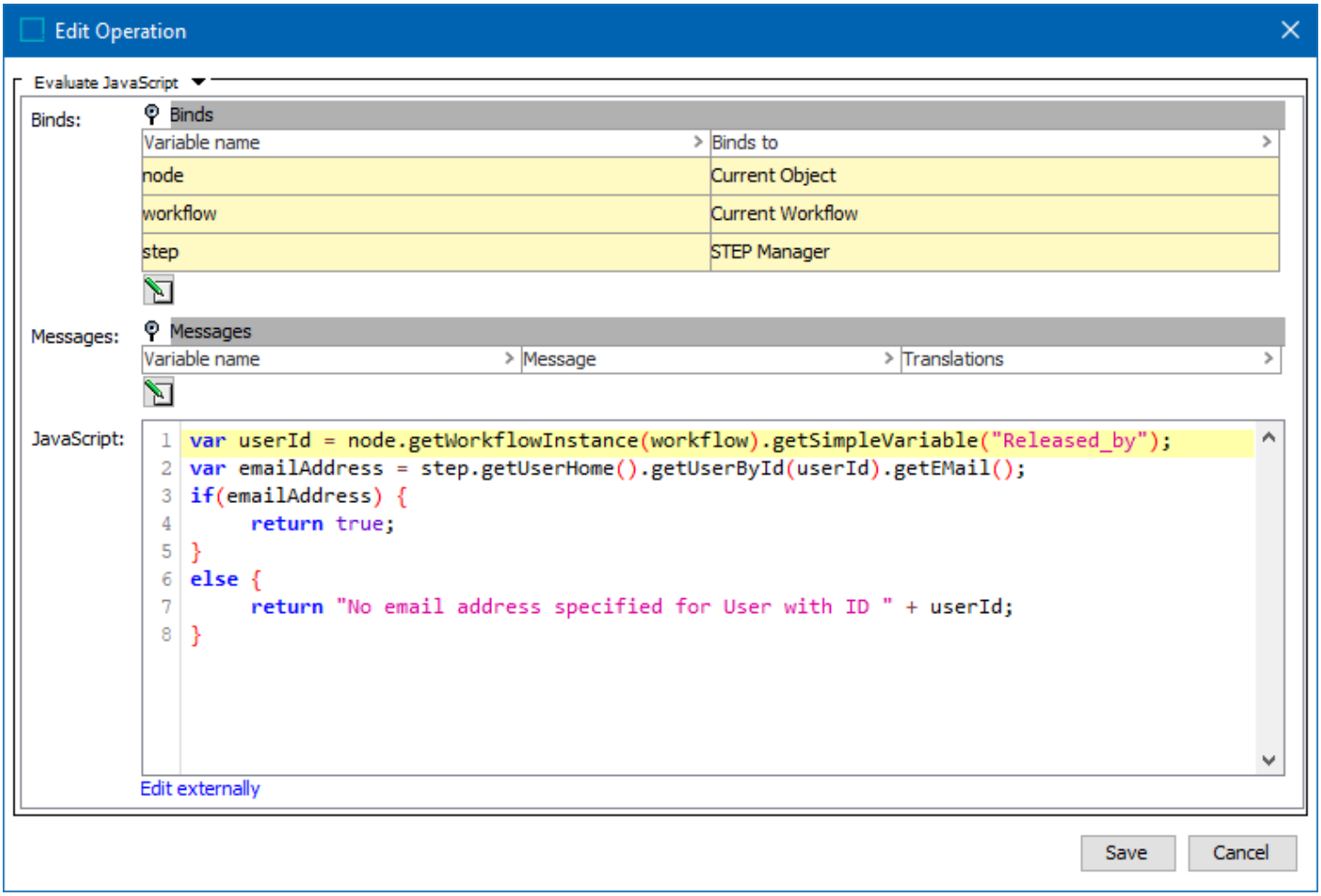
The business action script uses the variable 'Released_by' as well, where it is helping to identify who to send the workflow email to and notify that the task they just submitted was rejected. This script also is sending the email to the user without any attachments other than the message. Notice that the message shown in pink, can be crafted to say anything that might be helpful to the receiver whenever a task is rejected.



It is important to note that if there is no email address registered, or if the email is not a valid one, the executing user is not notified (e.g., no pop-up appears). However, it is captured in the STEP system log on the application server.

If the user doing the rejecting should be notified that no email was sent and provided with an error message, then a business condition must be added to the Transition Editor on the 'Condition' tab. The following script is what should be entered in below.

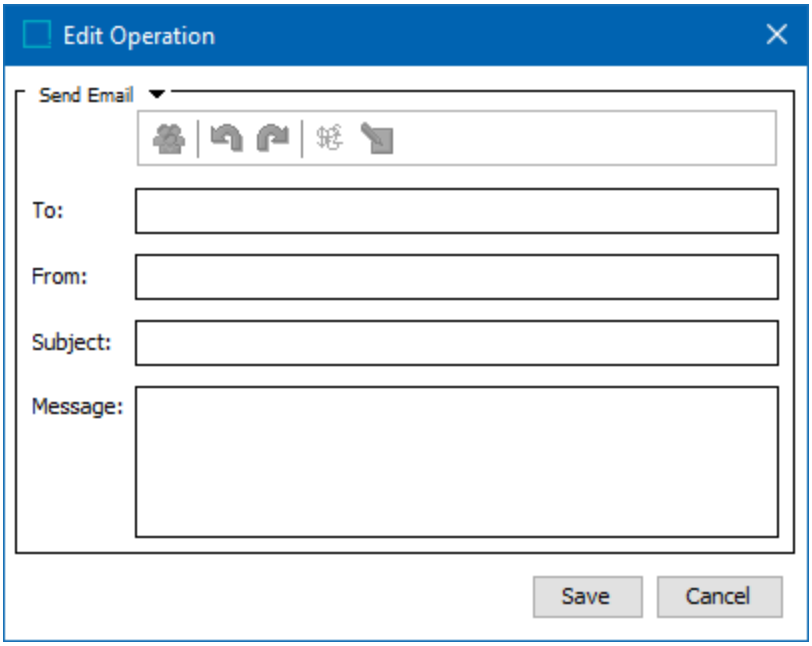
```
var userId = node.getWorkflowInstance(workflow).getSimpleVariable
("Released_by");

var emailAddress = step.getUserHome().getUserById(userId).getEMail();

if(emailAddress) {

return true;

}

else {

return "No email address specified for User with ID " + userId;

}
```



Static Email Address

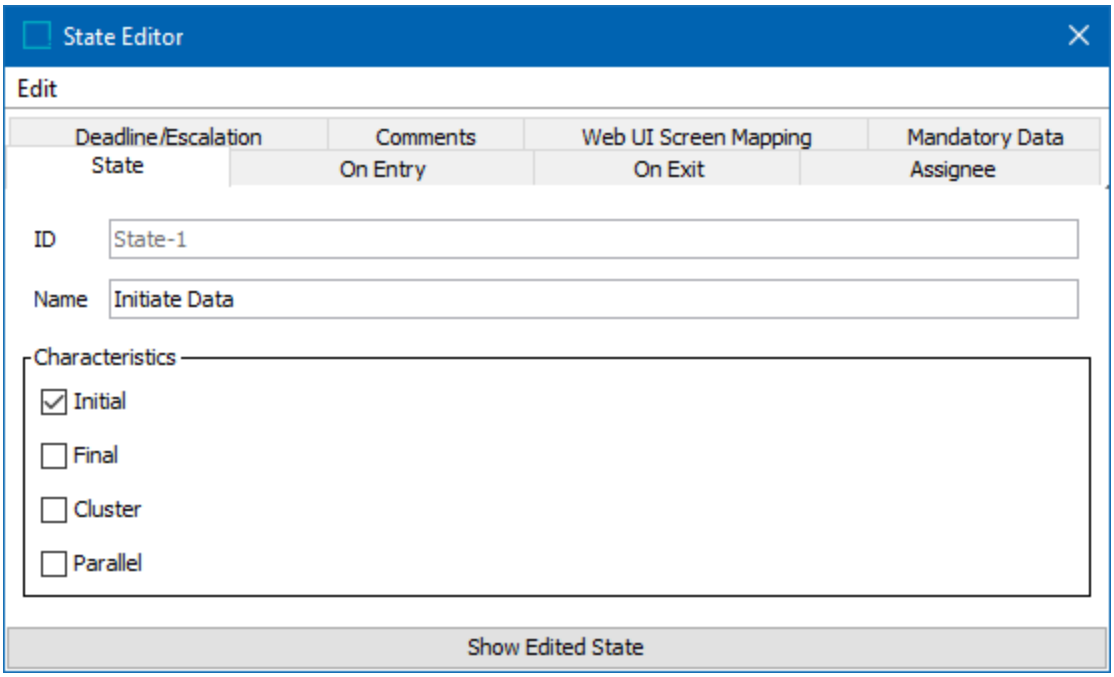
In the above scenario, STEP will trigger emails to the executing users. However, in some cases the same user will always need to be notified and never change (a static email address). In this case it is possible to send emails to the static email address with the 'Send Email' operation, and no JavaScript is needed.

This 'Send Email' operation can be added in a State Editor on the 'On Entry', 'On Exit' or 'Escalation' tabs. It can also be sent in the Transition Editor on the 'On Transition' tab. Select the appropriate Editor and tab and then select Add new Business Action > Merge into > **Send Email**.


Edit Operation
Send Email
To:
From:
Subject:
Message:
Save
Cancel

State Characteristics in Workflows

The State tab of the State Editor allows for workflow designers to assign specific characteristics to a state.



When creating states for a workflow, not all states need a characteristic assigned to them. Many states will not have any characteristics assigned, but in some cases state characteristics become very important. For example, in order for a workflow to be valid, it needs one Initial state and one Final state. Additionally, if clusters are present in a workflow, there also needs to be one Initial state inside the cluster.

This section of the documentation describes the various state characteristics and how they are used within workflows, specifically:

- Initial and Final
- Clusters
- Parallels

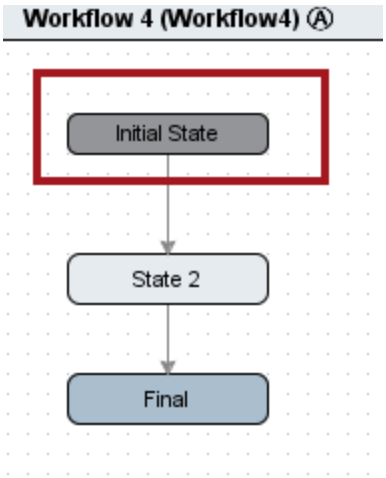
Initial and Final States in a Workflow

A workflow can have as many states as it needs in order to correctly depict the action or actions taking place. These states all must have at least one transition coming into the state and at least one outgoing transition from the state. However, the two states that every workflow need, the **Initial State** and a **Final State**, do not necessarily have to abide by these transition rules depending on the workflow layer they reside in.

Initial

Initial State

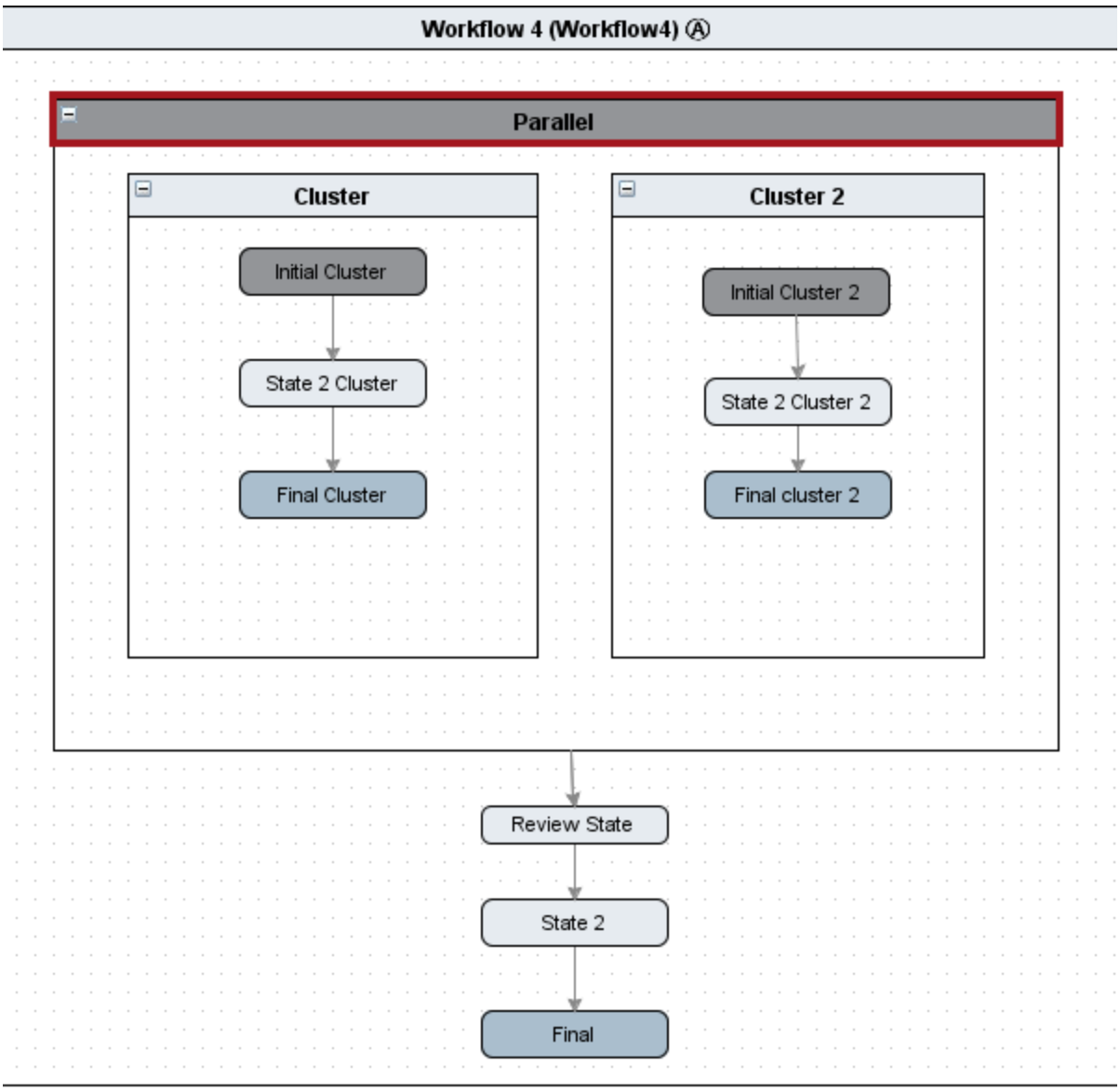
The **Initial State**, represented by a slightly darker colored state, signals to an object entering the workflow where to start. This could be in a simple workflow, such as the one pictured below, or it could be in regards to telling an object which state to start with when entering a cluster. There can only be one outer Initial State for the general workflow, and there can only be one Initial State inside each cluster. An outer layer Initial State does not have an incoming transition, only an outgoing transition. This outer Initial State signals to the object / task entering the workflow where to start. Once an object knows what state to start with, the workflow can progress.



For more on Clusters, see **Clusters in Workflows** in the **Workflows** documentation.

Initial as a Parallel

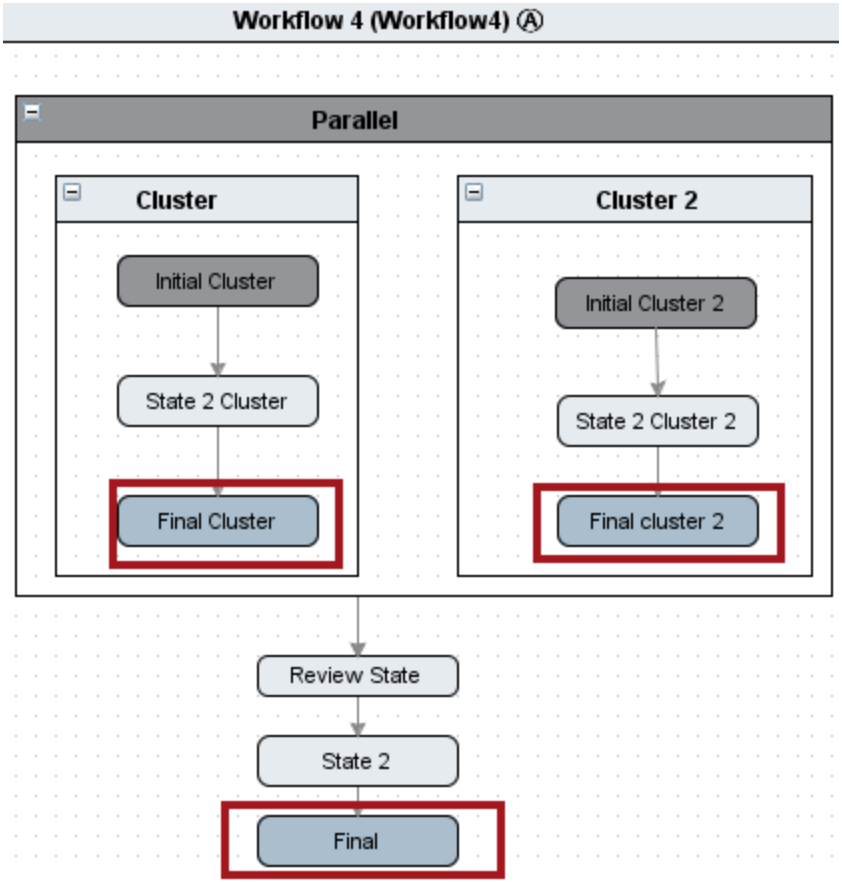
It is often necessary for more than one incident to take place at a time in a workflow. In this case, a **Parallel** structure should be used as the **Initial State**, which auto-initiates the object into the **Initial States** within each cluster simultaneously. The object only leaves the **Parallel** once the object has reached the **Final State** in each cluster.



For more on Parallels, see **Parallels in Workflows** in the **Workflows** documentation.

Final State

States with the **Final** characteristic are used to denote the ending of either an entire Workflow or a Cluster. Only Final States within Parallels and Clusters can have an incoming and outgoing transition.



All workflows should have a reachable outer layer **Final State**. Never-ending workflows can lead to performance impacts. However, there cannot be transitions out of a outer layer **Final State** characteristic, only incoming transitions. For this reason the outer layer **Final State** rarely represents a human task. When an object reaches a **Final State** on the outer level of a workflow (not in a Parallel or Cluster), it will remain there until the object is taken out of the workflow.

For performance reasons, and because the workflow cannot be started again on an object already in the workflow, it often makes sense to automatically remove objects from the workflow as they reach a **Final State** on the outer level.

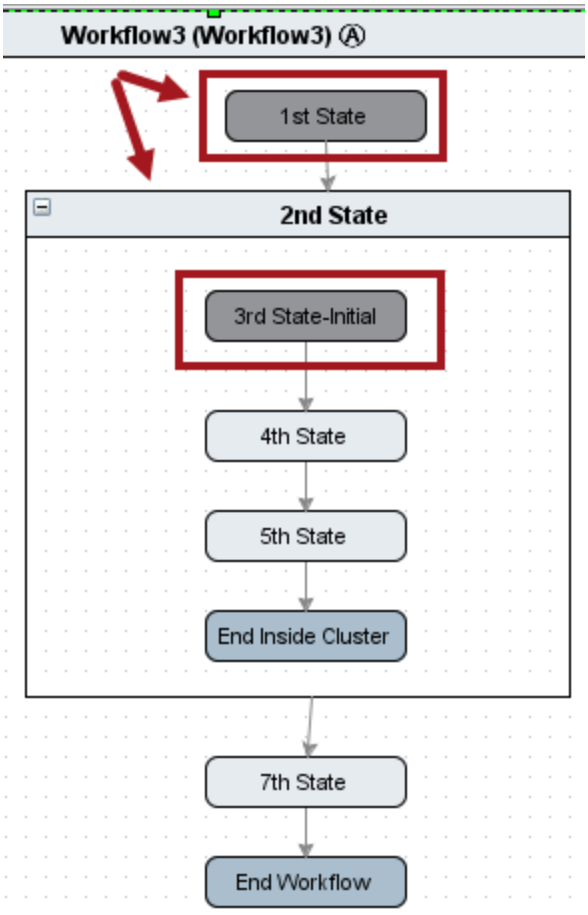
For more on how to automatically remove an object from a workflow see **Remove Objects from Workflows** in the **Workflows** documentation.

Clusters in Workflows

A cluster is a grouping of states. There are two main reasons why a grouping of particular states might be beneficial within a workflow. It may be useful for purely administrative reasons, making it possible to easily monitor a section in a workflow, or it may be used in conjunction with other clusters inside of a parallel, which is the more likely use case. For whatever the reason may be, it is important to know how to clusters work and how to create them.

Creating and Entering Clusters in Workflow

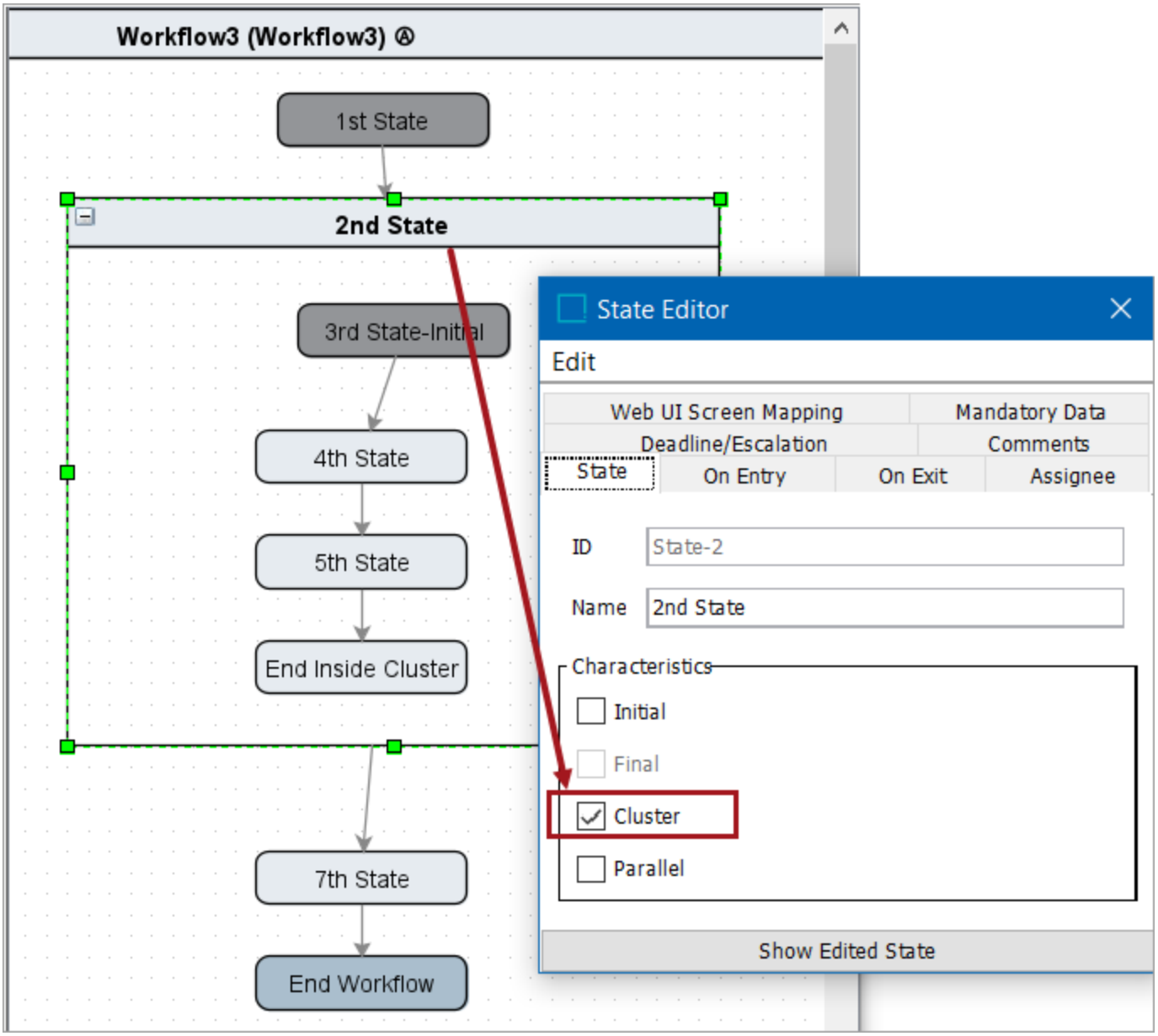
Any state can have the cluster characteristic added to it. This means that the particular state can have sub-states. The first sub-state in a cluster has to be an Initial state in order for the workflow to be valid. This means that there could be multiple states in a workflow with the characteristic of Initial state.



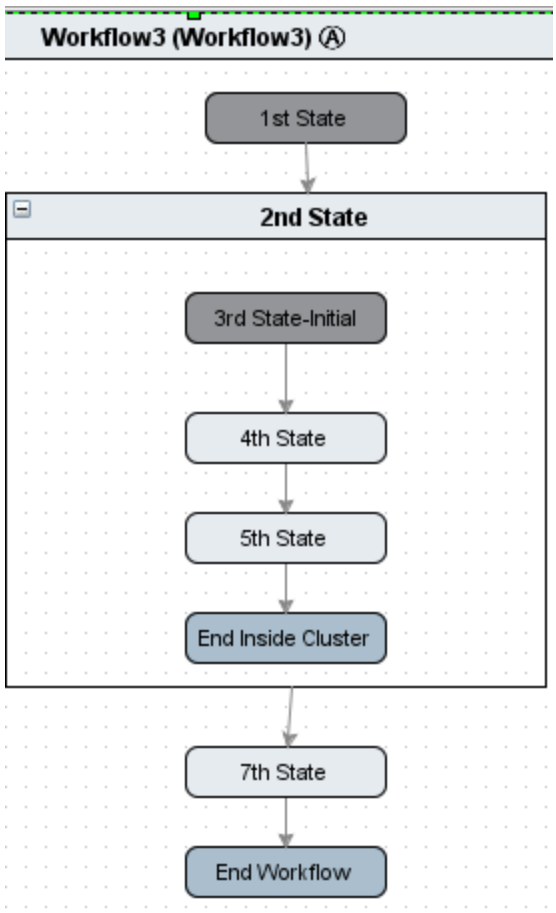
Note: For more on the Initial state, see **Initial and Final States in a Workflow** in the **Workflows** documentation.

It is important to note that states with cluster characteristics are still states in their own right. If an object is in a cluster sub-state, it is also at the same time in the state with the cluster characteristic.

If a state with a cluster characteristic contains sub-states, the cluster characteristic cannot be deselected in the State Editor.



In looking at the following cluster, it is important to know how an object placed into the workflow would move through it.

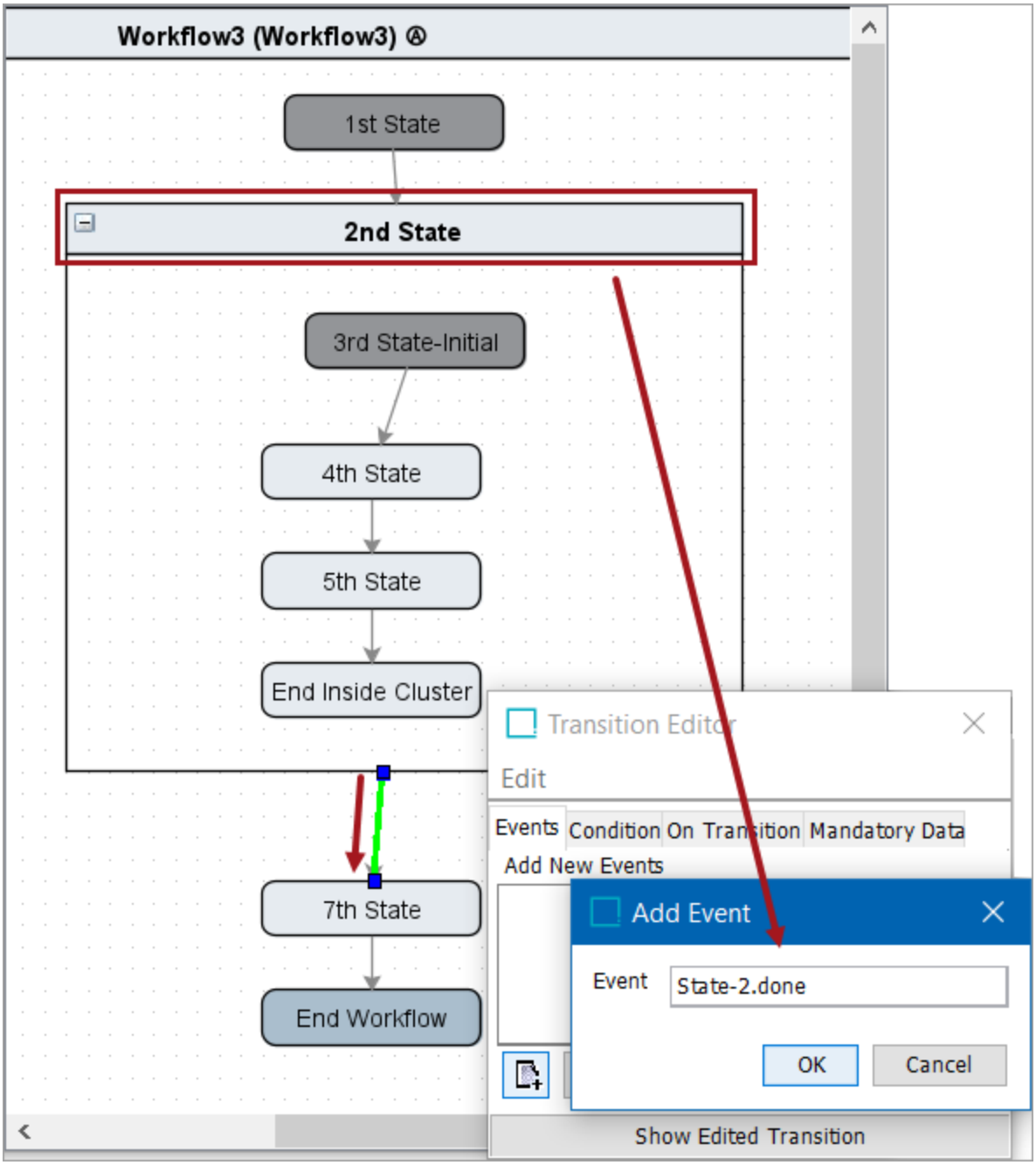


1. After entering into the **1st Initial State**, an object enters into the **2nd State,** which also happens to be a cluster.
2. Once in the cluster, the object also simultaneously enters into the **3rd State-Initial**.
3. It then proceeds through the next ordered steps until it reaches the **End Inside Cluster** (**Final state**). Because this is inside of a cluster, there is a transition allowed out of this **Final state** (which is different than what is allowed for the ultimate **Final state** of a workflow). In this case, it exits not only the **End Inside Cluster (Final state)**, but it also exits the entire cluster and goes to the **7th State**.
4. From the **7th State**, it proceeds to the **End of Workflow (Final state)**. The object has reached the end of the workflow.

Exiting a Cluster in a Workflow

When exiting a Final state in a cluster, the name of events on transitions can aid with the progression of the workflow. If an event on a transition has the naming formula of 'cluster state ID.done' the transition out of the Final state with the cluster characteristic to the next state will be triggered automatically.

For example, in the picture below, the name of the event on the transition from End Inside Cluster (which has the characteristic of a Final state) to 7th State is being named. Following the formula to trigger the transition automatically, the name takes on the 2nd State cluster's ID followed by '.done'.



This is useful for clusters that have their last state in the cluster with the characteristic as Final state, such as the one in the example. Normal functionality dictates that an object in a Final state would stay there until it is manually taken out or there is special coding set up to take the object out of the workflow automatically. However, with this naming system there is no need for a manual removal or special configurations to progress the object through the workflow. This naming configuration allows the object to leave the Final state within the cluster, and also leave the cluster state itself.

The system will always look for a transition out of a cluster that has an event name of 'Cluster ID.done'. If there is not a transition found with this name, the system will look for a transition out of the cluster with no named event and perform that transition.

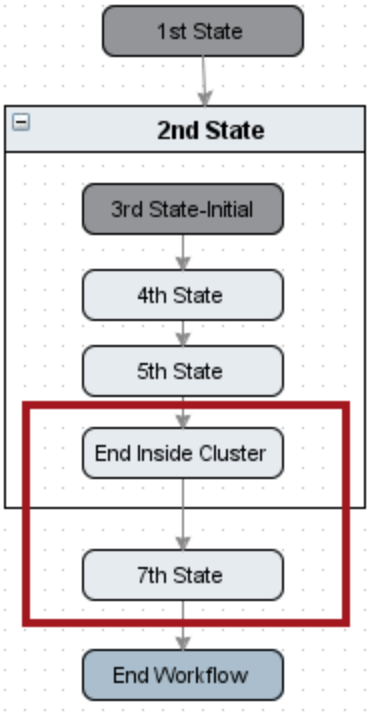
If there are two or more transitions out of the cluster that all have either the 'Cluster ID.done' event name or no named events, the system will randomly perform one of the transitions unless conditions on the transitions determine which one to perform.

If there are only transitions with named events different from 'Cluster ID.done' out of the cluster, objects will remain in the Final state and can only be progressed by triggering events on the cluster state.

Note: Prefixing an event on a transition with the ID of the state from which the transition goes and a dot ('.') will cause the event to be hidden to users working with the workflow in the workbench. This approach should therefore be used whenever you have a 'system' event that should not be triggerable by human users.

The image below shows an alternate way of configuring a cluster and transition relationship. There is no Final state inside the cluster. Instead there is a transition directly from End Inside Cluster to 7th State outside the cluster.

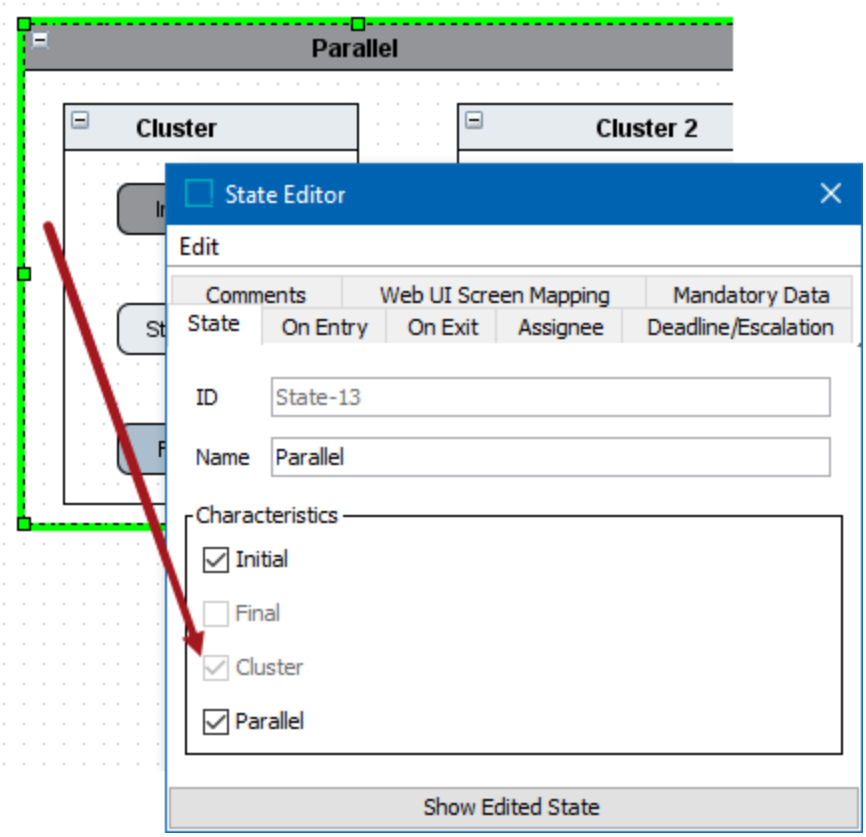
When this transition is performed, the object will exit both the End Inside Cluster state and the 2nd State cluster. While this configuration is perfectly valid, generally the first type of state setup with a Final state and transition is recommended.



Parallels in Workflows

Parallels

The Parallel characteristic is used for implementing two or more clusters simultaneously in the workflow. A Parallel state will automatically get the cluster characteristic indicating that it can only contain clusters as its immediate sub-states.



Once the Final states of all the clusters that are immediate children of a parallel have been reached, the parallel is automatically exited upon the next transition.

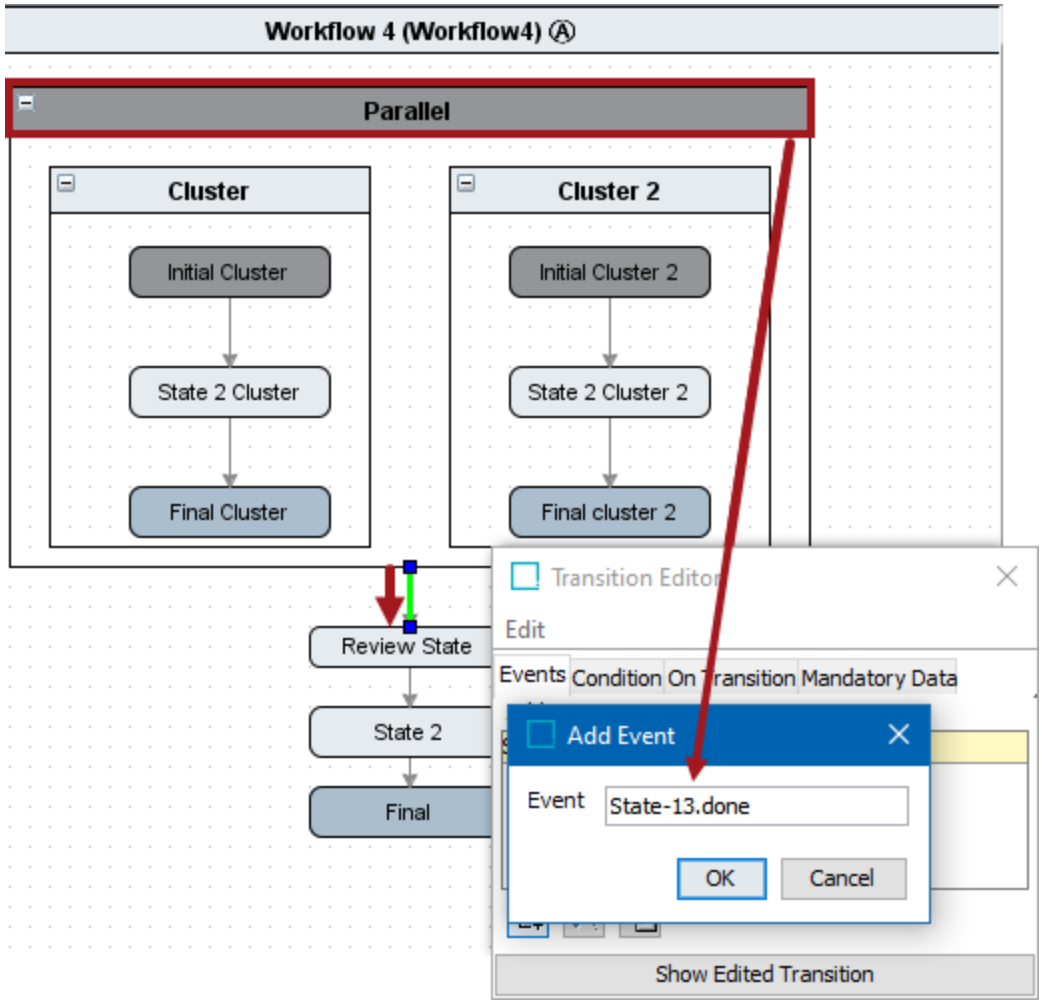
Exiting a Parallel

Transitions from parallels and Final states in clusters can be simultaneously and automatically triggered by using the formula Parallel ID.done to name the event. This then allows the object in the workflow to move to the next state outside of the parallel.

1. In the picture below, an object entering into the workflow would first enter into the parallel (as it is also the Initial state), and into the Initial states inside of each cluster simultaneously.
2. The object would then proceed through both clusters simultaneously until the Final state for each cluster is reached.

3. Once the object has reached the Final states in all clusters inside of the parallel, it then leaves the parallel and all clusters at the same time and enters into the first state outside of the parallel.
4. If the event on the transition out of the parallel is named with the formula of Parallel ID.done (in this case State-13.done), the object leaves both Final states in the clusters and the parallel at the same time, and automatically goes to the Review State which is situated outside of the parallel state. If the event is unnamed, or named without this formula, it would have to be manually moved to the Review State.

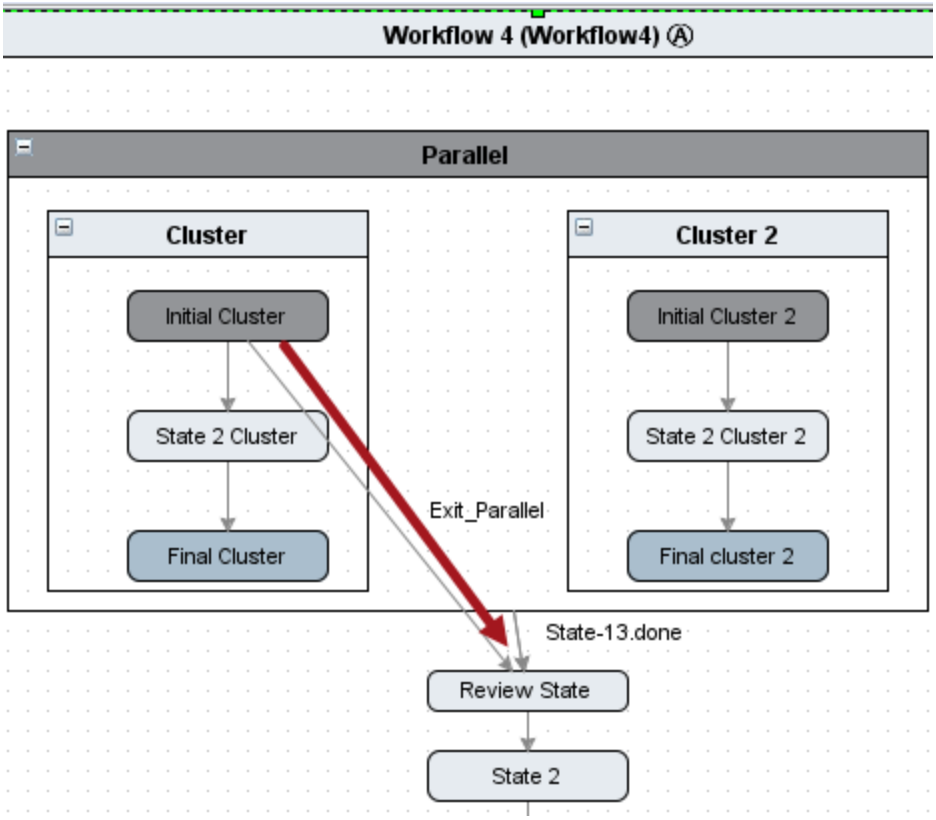
Note: When going through the workflow, it automatically will look for Parallel ID.done events first to be tried and performed, but if none exist it will then look for transitions with no special named events.



5. The object then continues through State 2 and ends in Final.
6. While the object is finished going through the workflow, it will not leave the workflow unless coding is in place to trigger it to leave, or the object is manually removed.

Note: Prefixing an event on a transition with the ID of the state from which the transition goes and a dot ('.') will cause the event to be hidden to users working with the workflow in the workbench. This approach should therefore be used whenever you have a 'system' event that should not be triggered by human users.

The image below shows an alternate way of configuring a parallel and transition relationship. There is a transition from Initial Cluster out of the parallel straight to Review State. If this transition is performed, the entire parallel and all its sub-states will be exited regardless of which state the object is in for Cluster 2.



Note: Parallel States are used for modeling concurrency and should only have child states with the Cluster Characteristic. Cluster and Parallel States do not typically represent human Tasks.

Status Flags

Status flags are an aspect of tasks in a workflow that can be useful to group tasks, and are often used to indicate tasks that need a higher priority or extra attention. For example, status flags can be used to signal tasks that have been returned to a previous state, tasks that are considered critical or time-sensitive, tasks that have errors, etc.

When a status flag is applied, it is specific to the task-in-state, similar to an assignee. An assignee does not travel with an object throughout the workflow, rather it is assigned at entry to each state, and upon exit from any state the current assignee is not retained (unless business rules are set up to record this information). Furthermore, when an object exits a workflow, there is no longer any assignee associated with it.

Status flags should be thought of in the same way as assignees. Configuring status flags requires the understanding that they are specific to a task in a state, and therefore must be applied at entry to any state. In addition, tasks in parallel states of a workflow may have different status flags applied in each state.

This section of the documentation describes how to configure status flags.

First, a place must be created to store the status flags, which is described in the **Initial Setup for Status Flags** topic.

Second, status flags must be configured for the particular workflow(s) in which they should be used, which is described in the **Configuring Status Flags** topic.

Initial Setup for Status Flags in Workflows

In order to use status flags for workflows, a setup group that can hold status flag objects must be configured. This only needs to be done once on any system and if already completed, this configuration is not required.

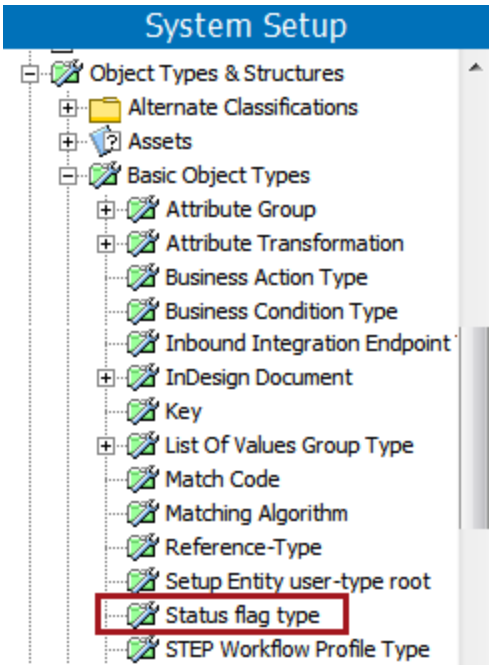
The required setup can be verified by skipping to step 3 below and checking if the option to create a status flag is available from the setup group that houses workflows or any other setup group defined as a parent on the 'Status flag type' defined in step 2 below. If you are able to create a status flag, then you may proceed directly to the **Configuring Status Flags in Workflows** topic to add status flags to a particular workflow.

Prerequisites

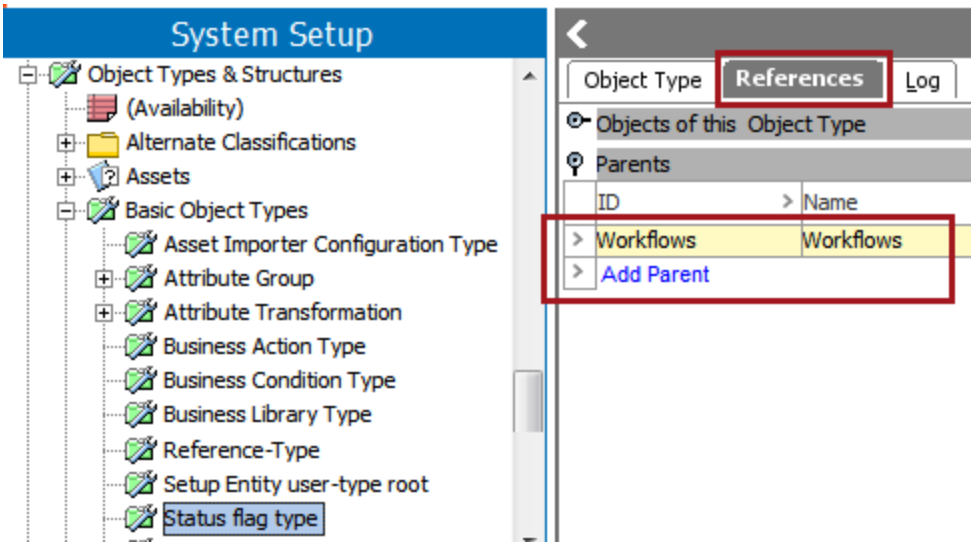
The below configuration instructions require that a setup group has first been created for workflows. If the Workflows setup group seen in step 3 below is not present, see the **Initial Setup for Workflows** topic to configure it.

Configuration

1. Go to System Setup and select **Object Types and Structures** > **Basic Object Types** to find the object type called **Status Flags type**.

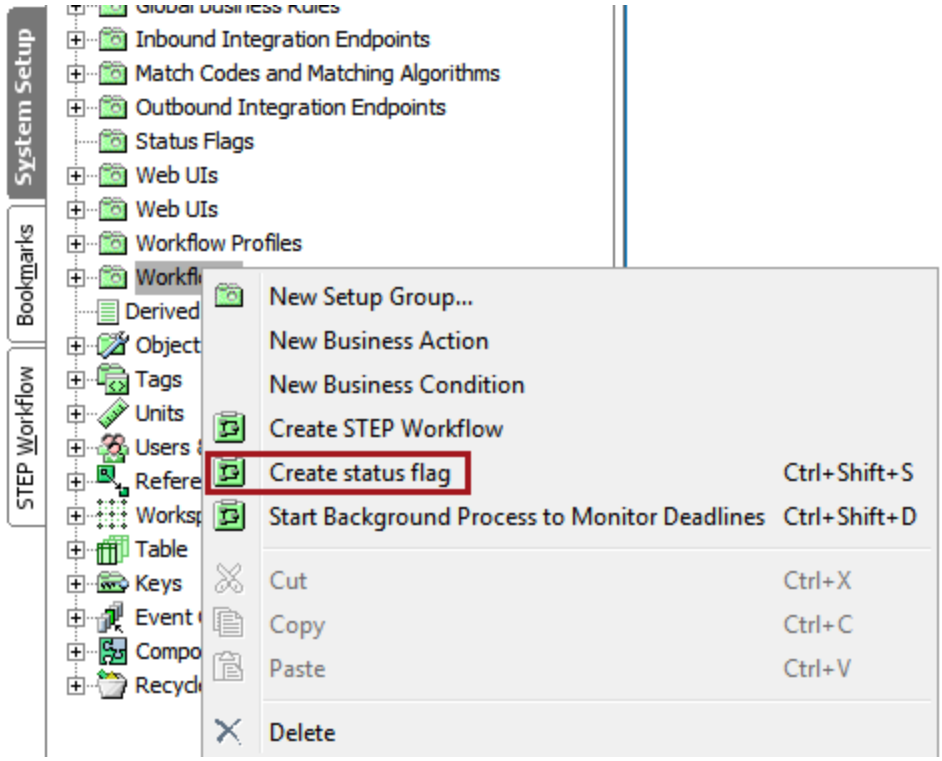


2. Select the **References** tab, click the **Add Parent** link, select the **Workflows** node (or any other node where status flags will be created) in the dialog that pops up, and click **Select** to close the dialog. The end result should appear as below:

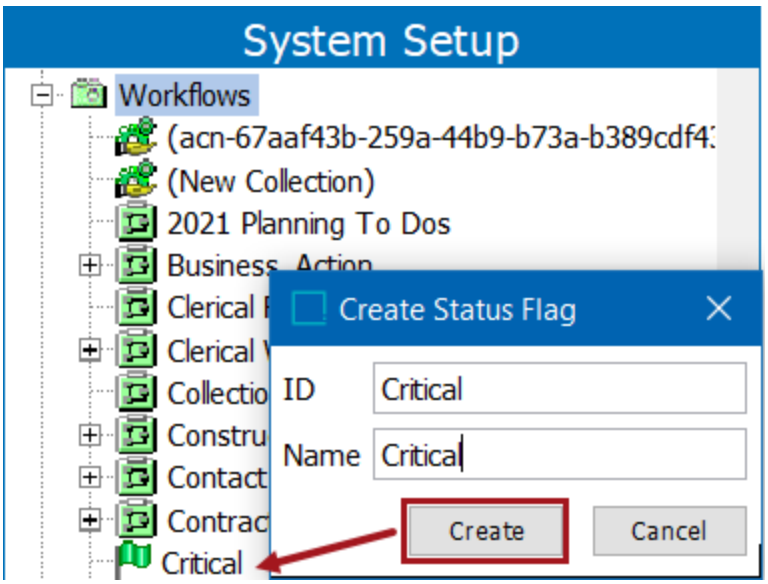


This allows status flags to be created under the selected setup group.

3. To create a status flag, go to the selected setup group, right-click on it, and select **Create status flag**.



4. Give it an ID, name, and click **Create**. Upon creation, the status flag will appear under the selected node.



Create as many status flags as are needed to support the workflows that require them.

Note: Any workflow using status flags will automatically have a default flag applied to each task. Therefore, it is useful to create a standard or 'Normal' flag that can be applied as the default.

Status flags are global, meaning that once they are created, they can be used by any workflows in the system. However, status flags are not available in any workflows until they have been applied to each workflow in which they are needed. See **Configuring Status Flags in Workflows** for details on how to do this.

Configuring Status Flags in Workflows

Status flags are global objects that must be applied to any workflow in which they should be used. Once the needed status flags are created under the top node that houses them, they can then be associated to the desired workflow.

This topic describes how to add status flags to workflows, and how to use business rules to set status flags on tasks.

Prerequisites

Before status flags can be added to workflows, a setup group must first exist that can hold status flags, and at least one status flag must have been created. For instructions on these processes, see **Initial Setup for Status Flags in Workflows**.

In addition, it is important to understand that status flags are specific to a task in a particular state. More information on this can be found in the introductory **Status Flags** topic.

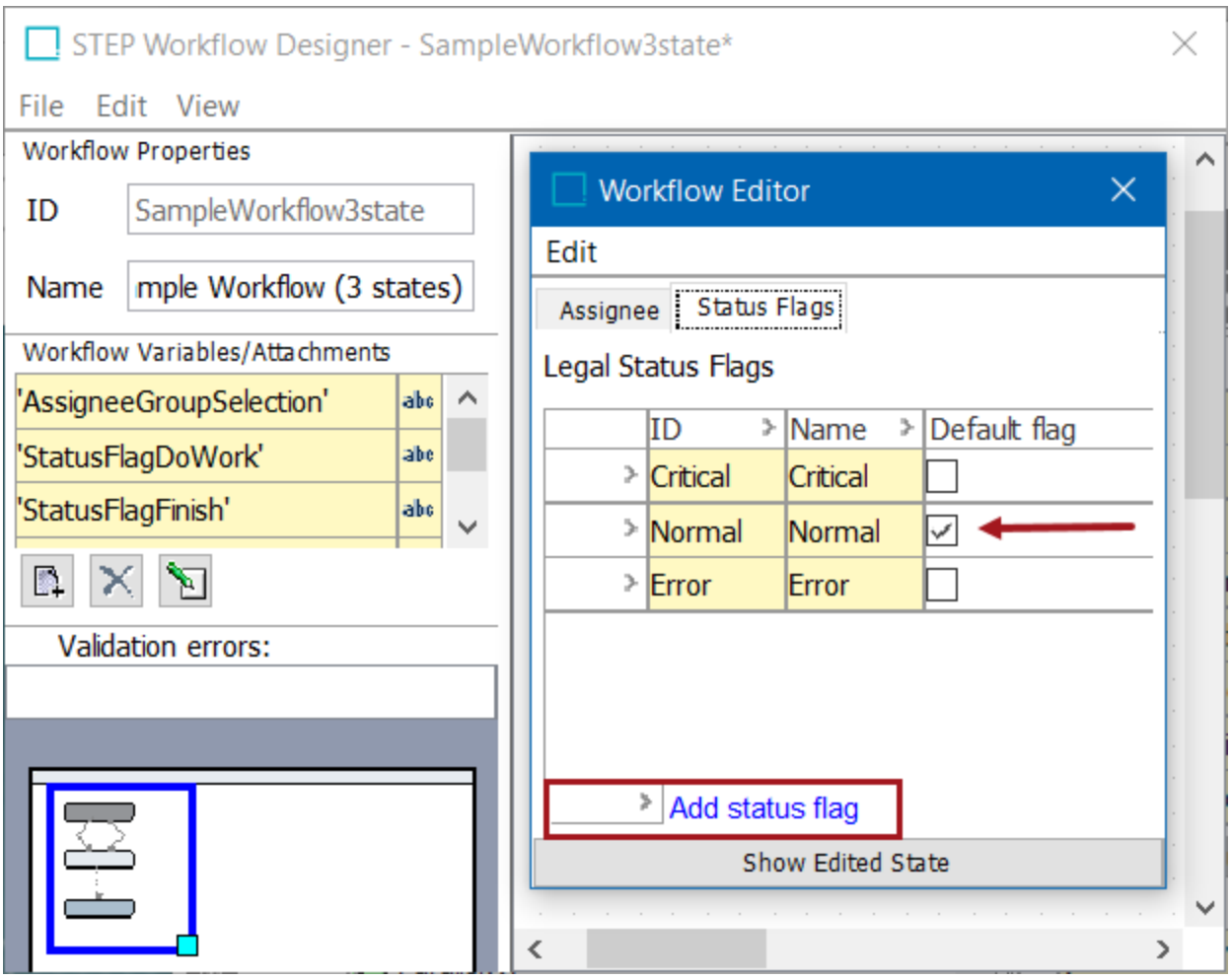
Configuration

Use the following steps to enable status flags for a workflow.

1. Open the **STEP Workflow Designer** of the workflow that needs status flags associated to it.
2. Double-click a blank area in the workflow frame, or right-click inside the workflow frame, and select **Edit Workflow** to display the **Workflow Editor** dialog.
3. Select the **Status Flags** tab, and add any flags that are needed for the workflow. Note that a 'default status flag' for the workflow must be selected.

   Important: Any new tasks assigned to the workflow will be given the default status flag unless the workflow is directed to do otherwise via business rules. Any existing tasks already in the workflow will also receive the default status flag.

4. When finished, close the editor, and save your changes.



5. Aside from the default flag applied to all tasks in the workflow, status flags are not truly enabled until business rules are created to apply them to tasks. Therefore, instructions for using business rules to apply status flags are included below.

Using Business Rules to Set Status Flags in Workflows

While it is not required to use workflow variables in conjunction with status flags, it is often useful to create workflow variables to be associated with each state in which a status flag should be applied. This allows for greater flexibility in how and when status flags can be set. This is because status flags are somewhat like workflow assignees in that they are specific to an object in a given state, and cannot be set in advance. Just as you cannot assign a task to a user until it enters the applicable state, you also cannot bind a status flag to an object's presence in a workflow until it enters the state. As it is often something from a previous state that determines what the status of the object is in a subsequent state, workflow variables can be used to capture and store what the flag should be, which can then be set on entry to the appropriate state.

The below instructions describe how to set a status flag based on a transition being taken. For example, a task that has been rejected may need to be marked as 'Critical' as it is now spending additional time in the workflow. This is not the only way to work with status flags, but serves merely as one example of one common use case. Rather

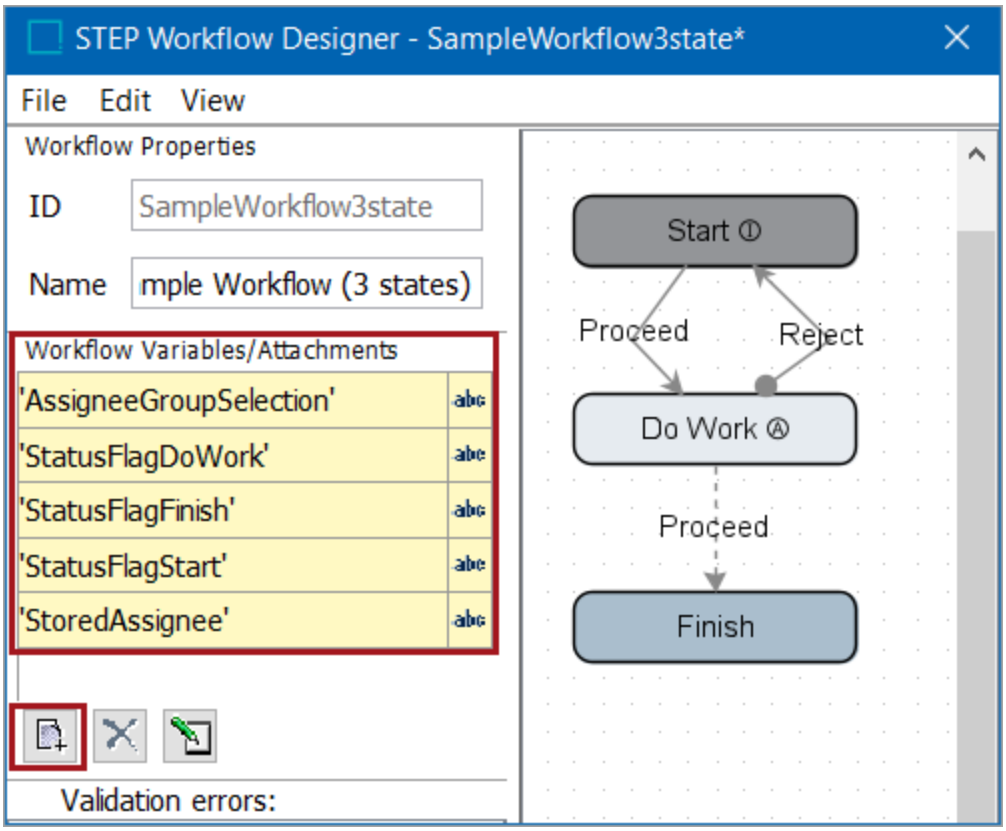
than using a transition to determine the status flag for a task, any data could be used, such as an attribute value, reference value, presence or absence of particular data, etc. The options for setting status flags are limited only to what can be done within business rules.

However, the options for end users viewing and accessing status flags are limited by the interface in which they access workflows. If workflows are accessed in Web UI, it is not required that status flags be stored in workflow variables. However, if end users will interact with the workflow only in the STEP Workbench, status flag assignments must be stored in workflow variables as that is the only way to make them visible to the end user.

Use a Workflow Variable to Capture a Status and Apply a Status Flag

The below instructions walk through an example of how to capture a desired status flag based on a particular transition being taken, store the information in a workflow variable, and subsequently apply the desired status flag at entry to a subsequent state.

1. Create a workflow variable to hold the status flag using the button. As status flags are state-specific, it is recommended to create one workflow variable for each state in the workflow that will have a status flag determined by some factor other than entering the state, so that when the determining factor is evaluated, the resulting flag to be applied at any given state can be stored for that state.

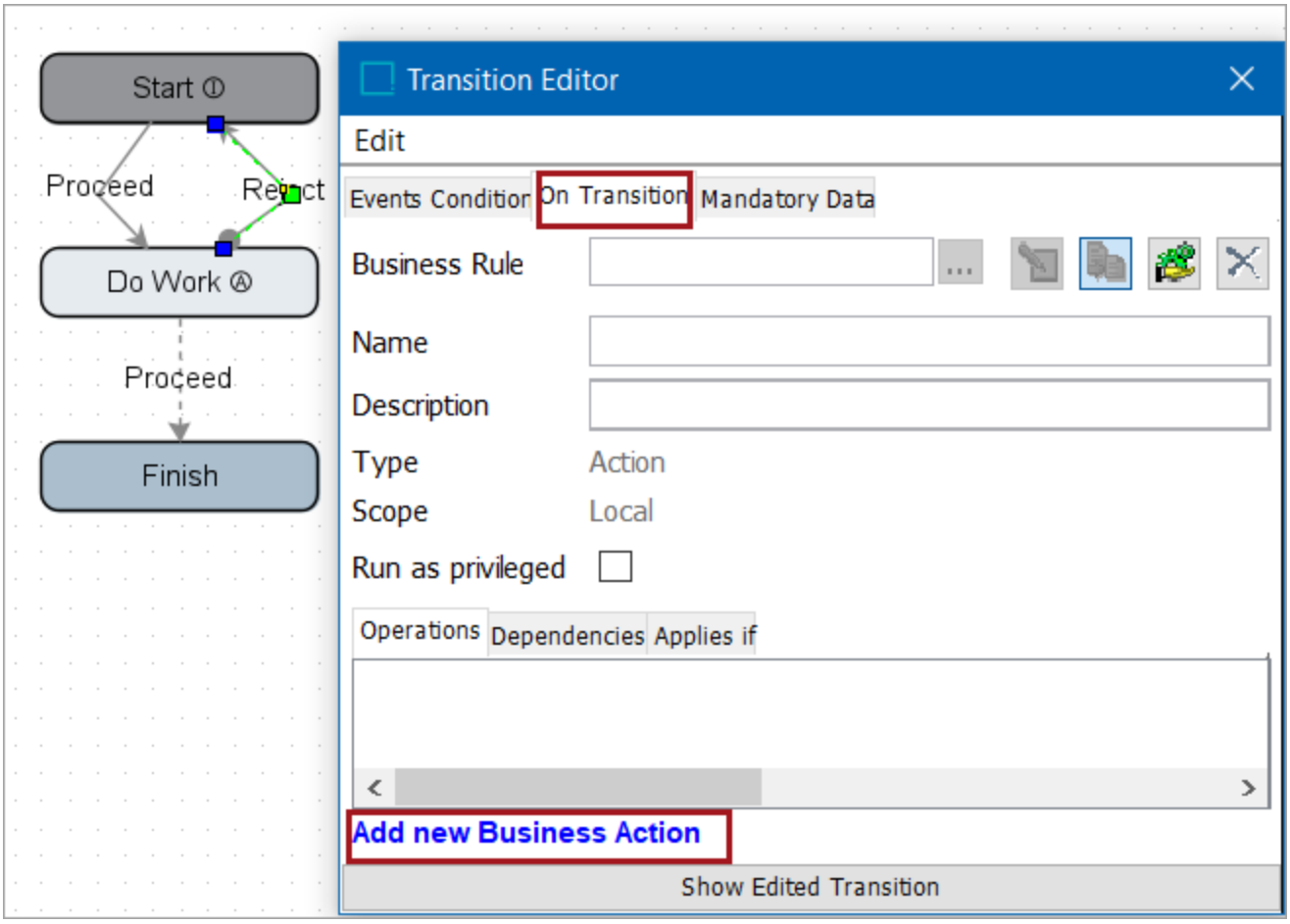


   For additional information on workflow variables, see the **Workflow Variables** section of the documentation.

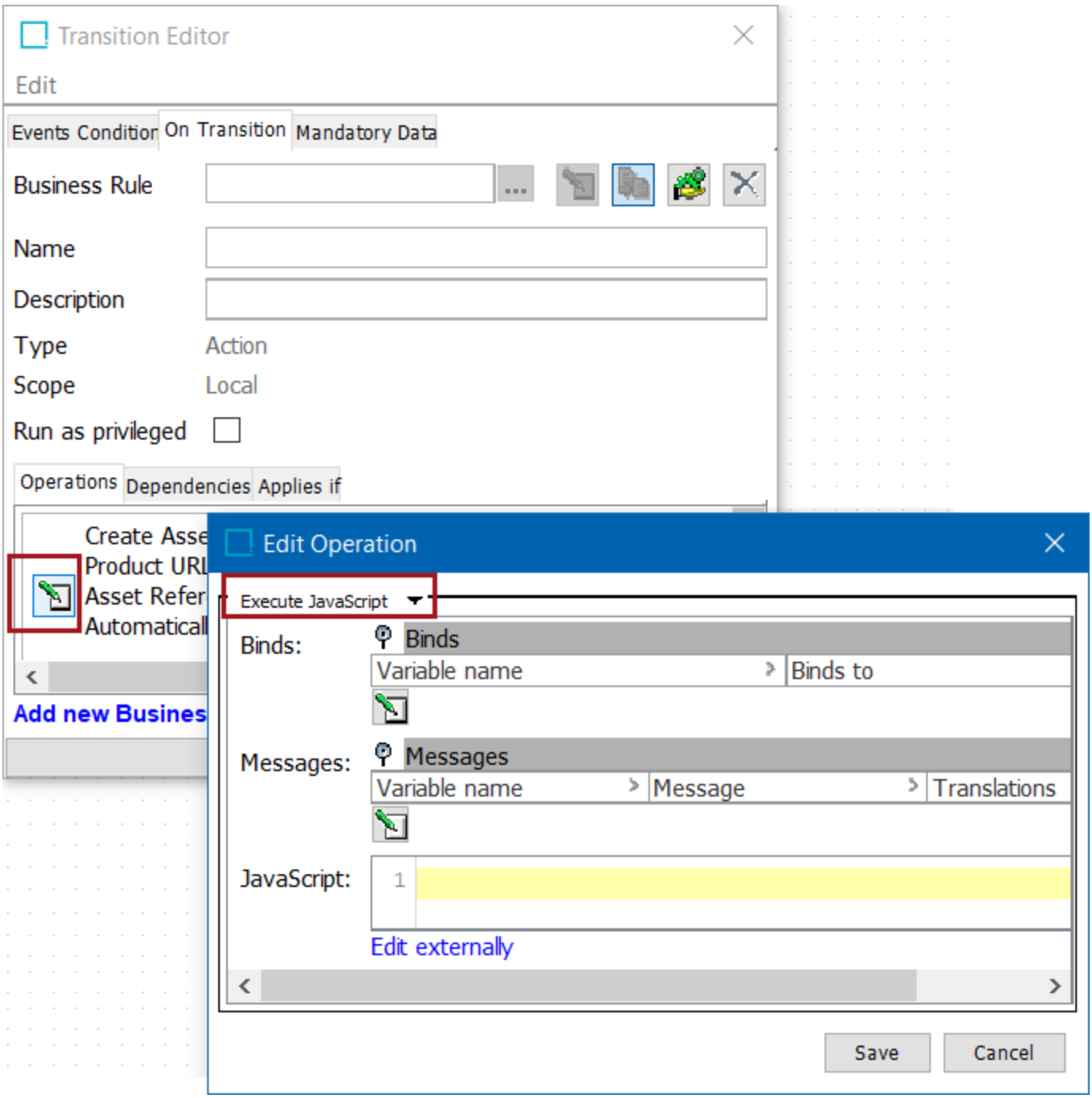
2. For this example, when a task is rejected, we want to apply a Critical status flag so that users know to quickly process the task in the Start state. Therefore, we need a business rule to record that the Reject transition has

been taken and store that information. As we want the task to then have a Critical status in the Start state, we will store the information in the StatusFlag_Start workflow variable.

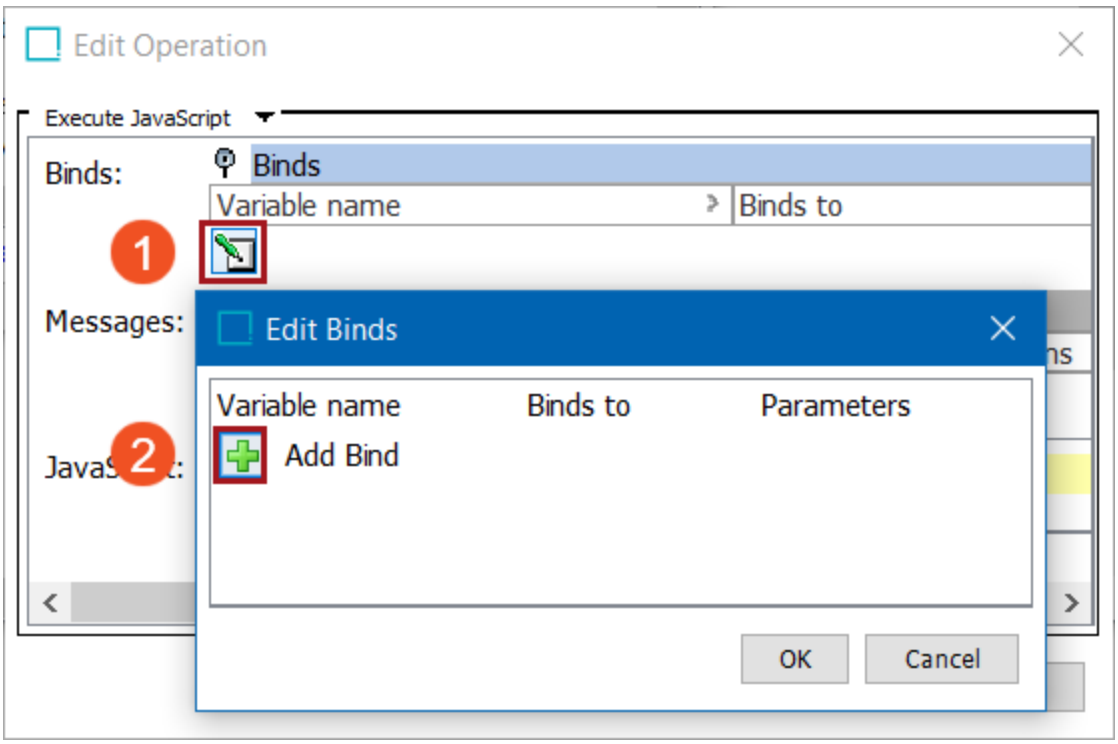
Right-click on the transition that should be recorded, select **Edit Transition**, click on the **On Transition** tab, and click the **Add new Business Action** link.



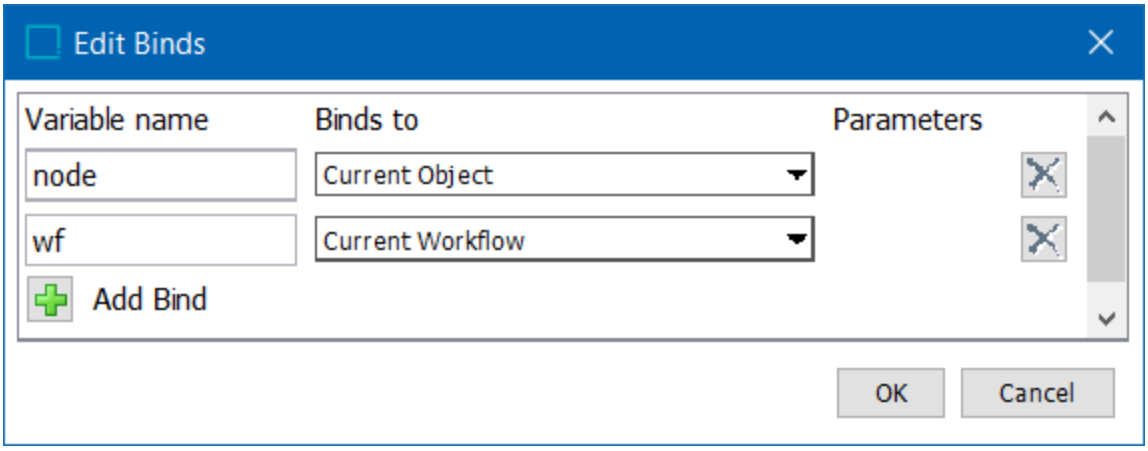
3. Click on the **Edit** icon button that appears under the **Operations** tab. This will bring up the **Edit Operation** dialog. Select the **Execute JavaScript** dropdown option.



4. Under Binds, click the **Edit Binds** button, and then click the **Add Bind** button.

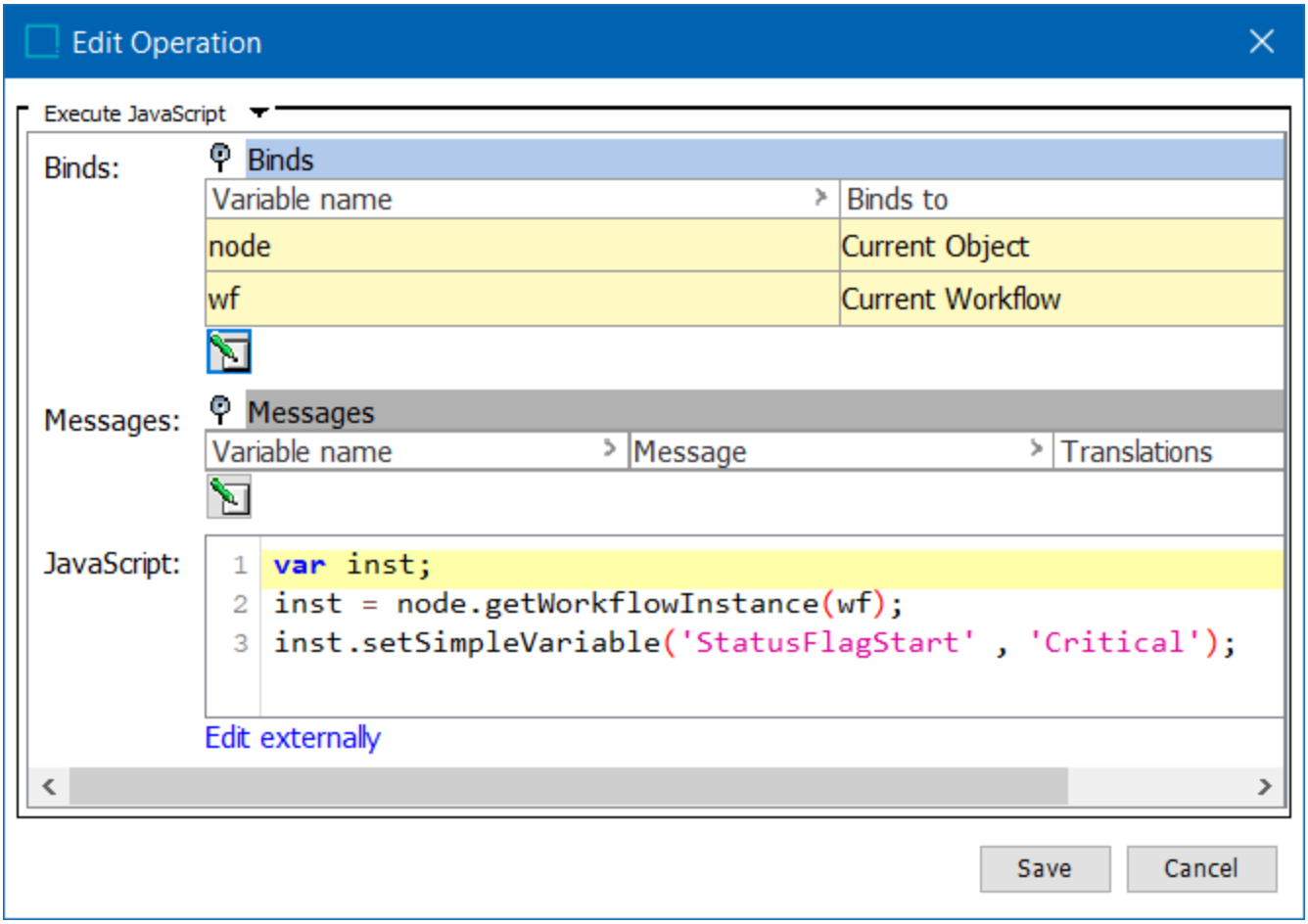


5. For this example, we need to bind to both current object and current workflow. Input variable names and select the appropriate binds, as shown below. Click OK to close the Edit Binds dialog when complete.



6. In the JavaScript field type in the appropriate JavaScript formula. An example is provided below that will set the value of the workflow variable 'StatusFlag_Start' to 'Critical'. Keep in mind that what is written in the parentheses is driven by the variable name that was written in the bind, the workflow variable created to hold the status flag value, and the ID of the status flag that is to be assigned.

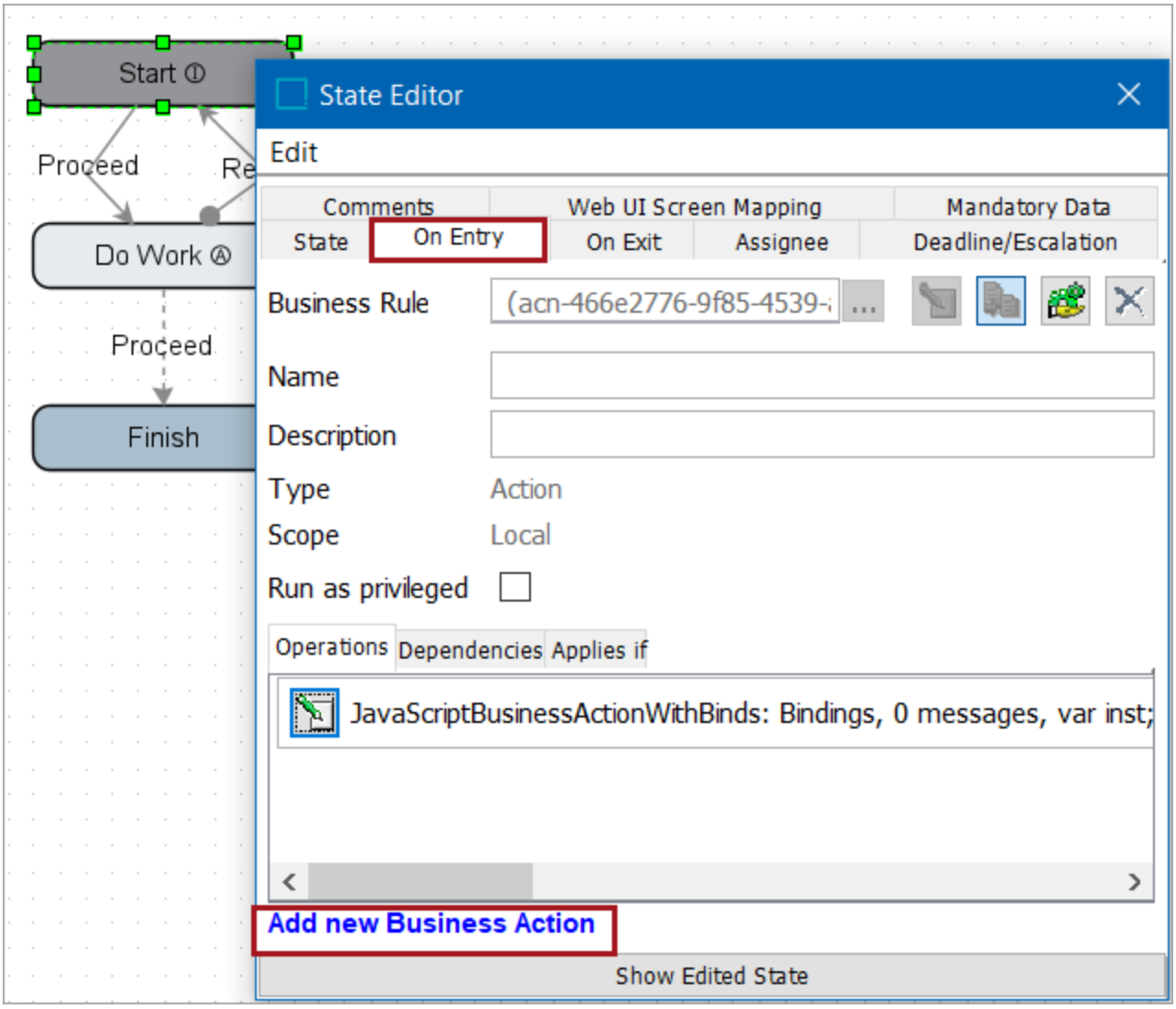
```
var inst;
inst = node.getWorkflowInstance(wf);
inst.setSimpleVariable('StatusFlagStart','Critical');
```



Click **Save** in the Edit Operation dialog when finished, and close the Transition Editor using the **X** button in the upper right corner.

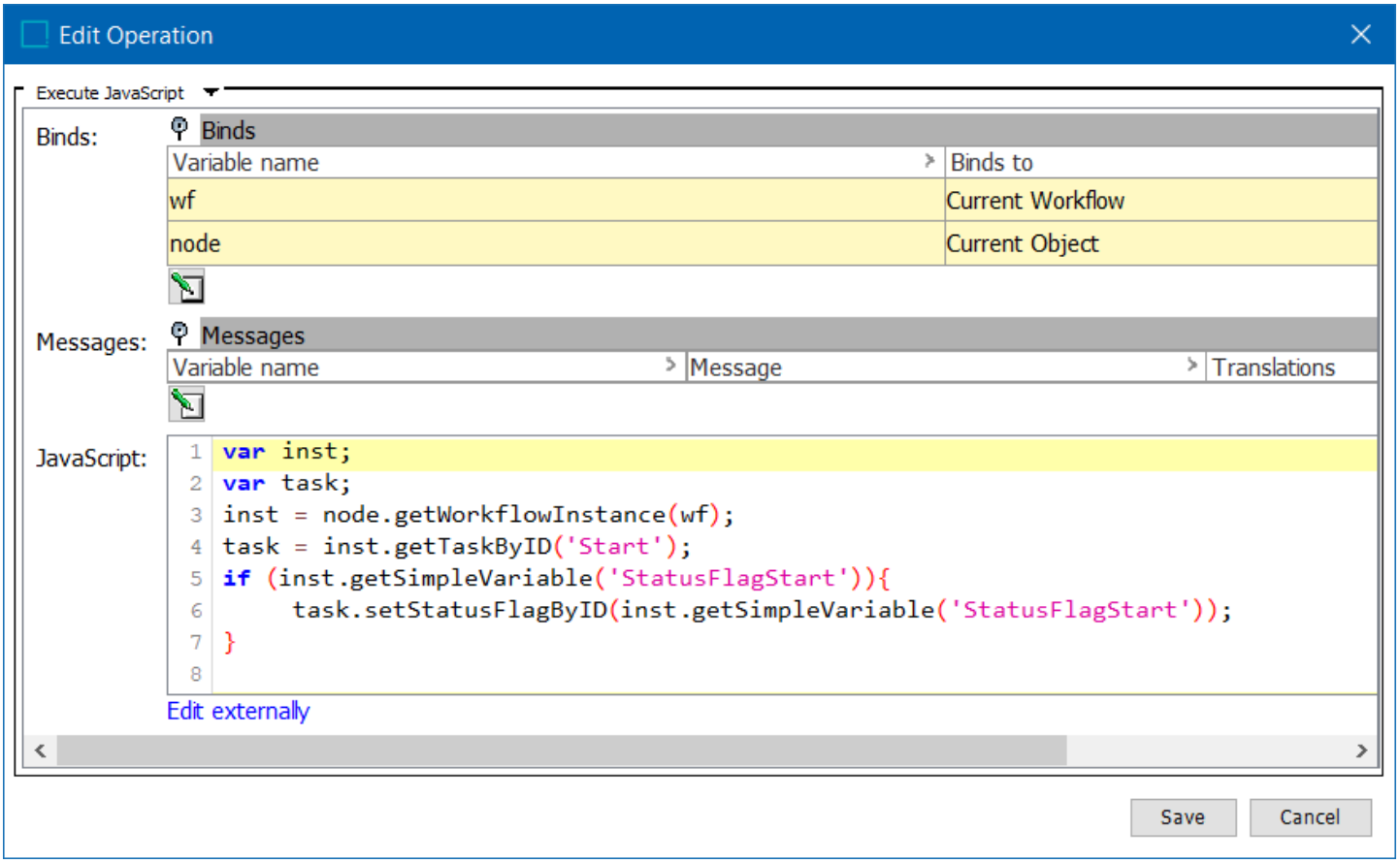
7. At this point we have only stored what the status flag should be, but have not applied the status flag to any task. To do that, a rule must be added to the state in which the flag should be assigned.

   Right-click on the state and select Edit State. Click on the **On Entry** tab, and click the **Add new Business Action** link.



8. Repeat steps 3 - 5 to edit the rule and add the Current Object and Current Workflow binds.
9. Enter the appropriate JavaScript formula. An example is provided below that will take the value of the previously stored workflow variable and apply it to the task entering the state.

```
var inst;
var task;
inst = node.getWorkflowInstance(wf);
task = inst.getTaskByID('Start');
if (inst.getSimpleVariable('StatusFlagStart')){
    task.setStatusFlagByID(inst.getSimpleVariable('StatusFlagStart'));
}
```



Click **Save** in the Edit Operation dialog when finished, and close the State Editor using the **X** button in the upper right corner.

At this point, rules have been created to capture the desired value for a status flag, and to set it upon entry to a state. However, in order to make the status flag available for end users to view when working with tasks, additional steps must be taken.

- If the user will interact with tasks in the workbench, the workbench view must be configured to include the workflow variable holding the status flag value. Additional information on configuring views is available in the **Views and Mappings for Workflows** topic in the **Getting Started with STEP Workflows** documentation.
- If users will interact with workflows in Web UI, the appropriate status flag components must be added to the screen. Additional information on working with status flags in Web UI is available in the **Status Flags in Web UI** topic in the **Web User Interfaces** documentation.

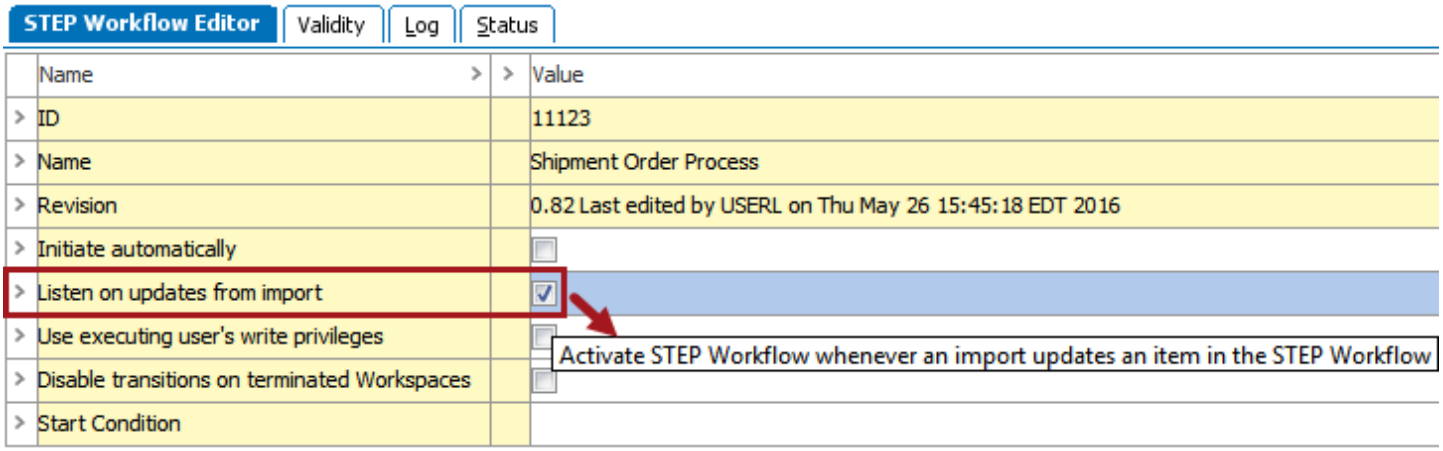
Triggering Workflow Events from Imports

If a workflow is valid for either products, classifications, assets, or entities, it is possible to configure the workflow to automatically trigger certain transitions if an object in the workflow has an attribute value changed in an import as long as the event on the transition has the ID of 'sibo.import'. This is helpful in that a user does not have to recheck any existing items that are already in the desired workflow, nor do they have to take the objects out to be updated, only to put them back into the workflow again. Instead, the objects already initiated into the workflow will be updated and then be triggered to move to the next sate, if the event on the transition has the proper ID.

For example, a user has Object A in their desired workflow in the Initial state. The Initial state has an event on the transition coming out of it with the ID of 'stibo.import' going to State 2. The user updates and imports objects, one of which is Object A. Due to the setup of the Initial state, Object A is updated, and an event is triggered moving Object A to State 2.

Condition for Triggering Workflow Events from Imports

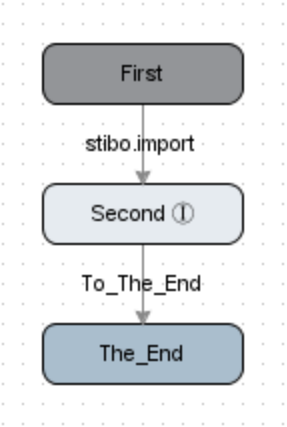
To turn on this functionality for a workflow, select the desired workflow in System Setup and check the **Listen on updates from import** checkbox on the STEP Workflow Editor tab.



Make sure that any events that should occur have the proper ID. Again, the Import Manager can only trigger events with the ID "stibo.import". Thus to have a transition performed for an object in a workflow when it is changed in an import, the object must be in a state out of which there is a transition with the Event "stibo.import" (the "stibo.import" event will not be visible in the STEP Workflow Items editor or on the Object Tasks tab).

Note: It is important to notice that only changes to attribute values (add, modify, remove) will trigger the "stibo.import" event. Thus, changing a reference, re-parenting an object or similar will not cause the event.

The image below shows a simple workflow with the functionality turned on.



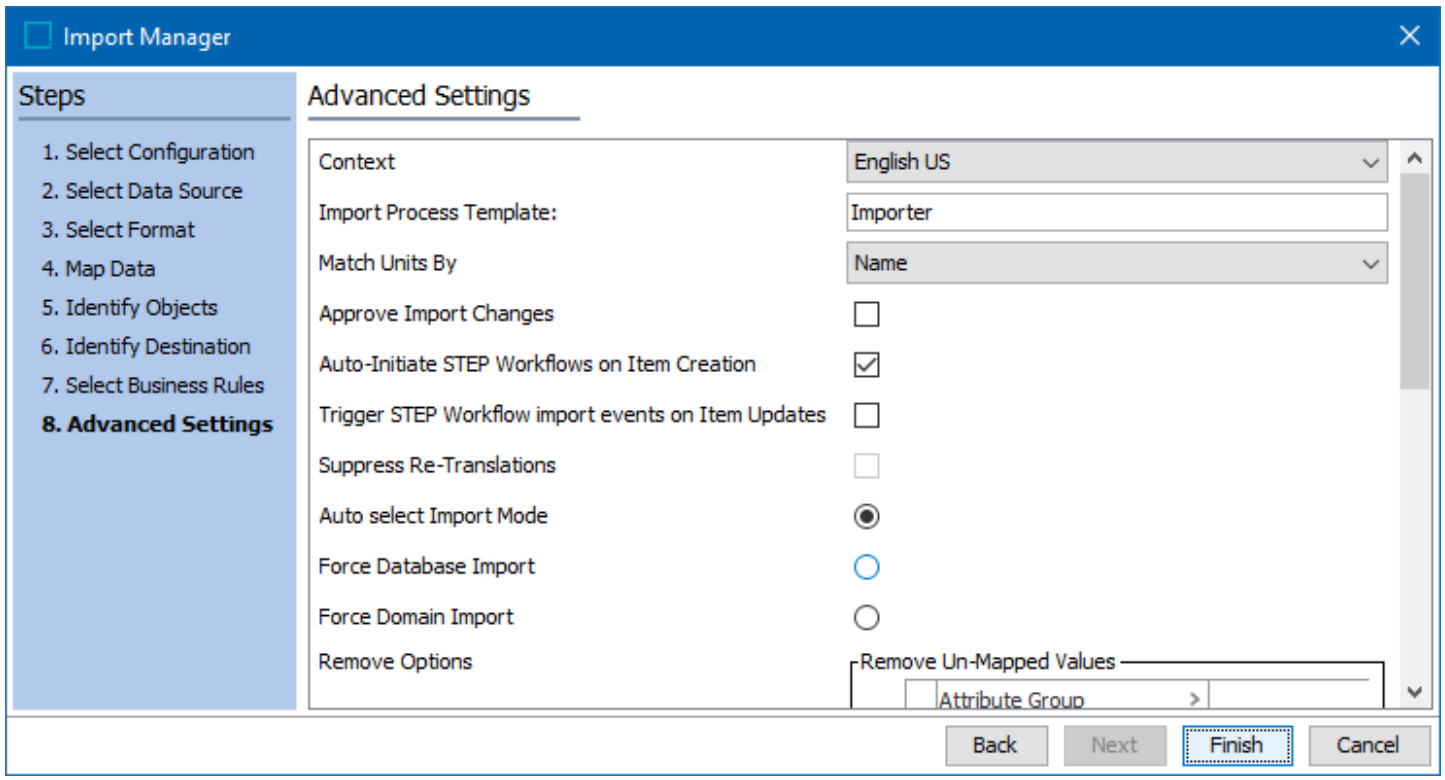
The "stibo.import" event is on the transition from the First state to the Second state and thus, if an object is in the First state and an attribute value is changed in an import, the transition will be performed.

When working with events generated by imports, there is a special reserved variable named "modifiedAttributes" available in JavaScript. This variable will hold a list of IDs of the attributes changed in the import. In looking at the example above, the variable will be available when leaving the 'First' state (On Exit), on the transition (On Transition + Condition) and when entering the state to which the transition goes (On Entry on the Second state).

Using this reserved variable a user could for instance add a condition on the transition only allowing the transition if a specific attribute has been changed and has a value after the import. This could be done with a script as follows:

```
var attChanged = false;
var modArray = modifiedAttributes.toArray();
for(var i = 0; i < modArray.length; i++) {
        if(modArray[i] == "ID of Attribute") {
                attChanged = true;
                break;
        }
}
if(attChanged && node.getValue("ID of Attribute").getSimpleValue()) {
        condition = true;
}
else {
        condition = false;
}
```

It is optional for users running imports whether or not the import potentially should trigger events. Thus for column based imports, in the Import Manger on the Advanced Settings page, the checkbox 'Trigger STEP Workflow import events on Item Updates' controls whether or not events should be triggered.



In XML, the functionality is enabled / disabled using the STEPProductInformation tag attribute "STEPWorkflowImportEvent" that can be set to "Y" or "N".

Since it is optional for users whether events should be triggered, the functionality is typically used in imports based on pre-configured import configurations. A use case could be one where an object is in a state waiting for data to arrive from an external system via an Inbound Integration Endpoint.

Note: It is only possible to trigger events from imports when running in Domain mode.

Validate Data in a Workflow

It is common for workflows to require some basic data validation to ensure that quality standards are met for the provided data. Data validation can be easily accomplished in a workflow by applying conditions on transitions.

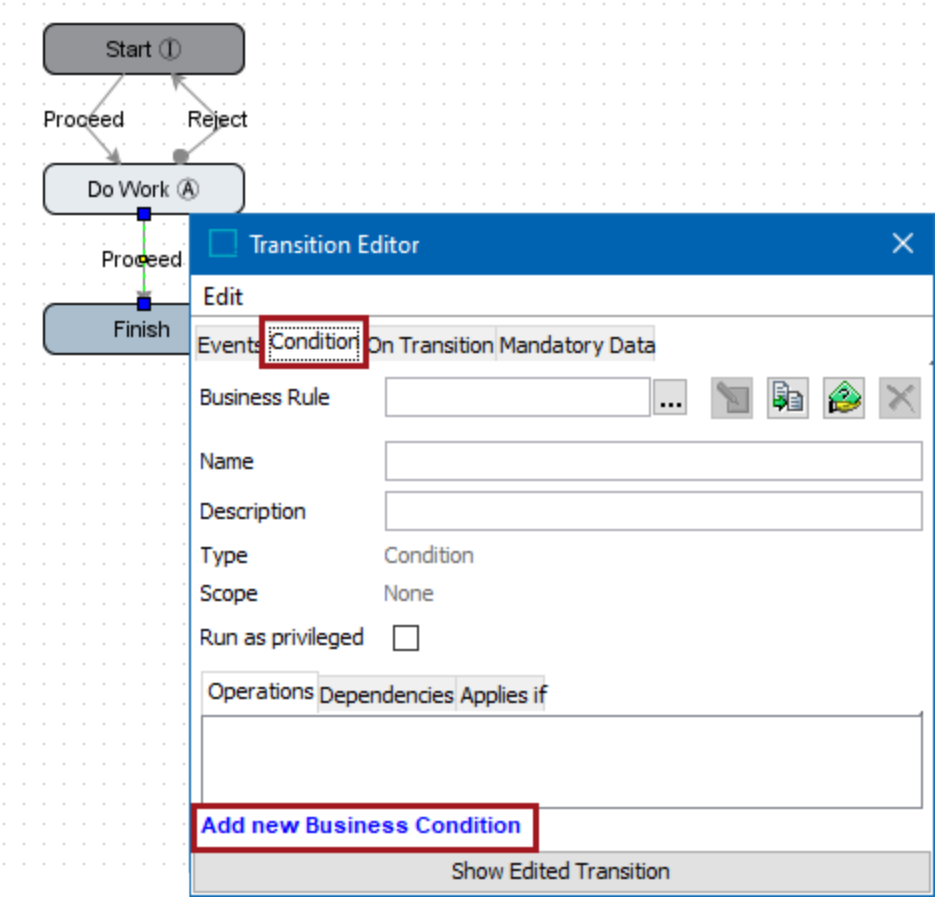
Business conditions will always evaluate to either true or false. In workflows they can be applied on transitions, and thus determine whether or not a transition can be performed. If a condition evaluates to true, the task may take the transition. If a condition evaluates to false, the task cannot take the transition and must attempt another transition or remain in the state until the data has been corrected and the condition evaluates to true.

When using conditions for basic data validation, the transitions are generally applied on user-accessed states, so that the user sees an error when attempting to submit via a particular transition if the data does not validate. Note that conditions should not be used to validate that mandatory attributes are populated as that can be done using the built-in mandatory attribute functionality, described in the **Mandatory Attributes and References in Workflows** topic.

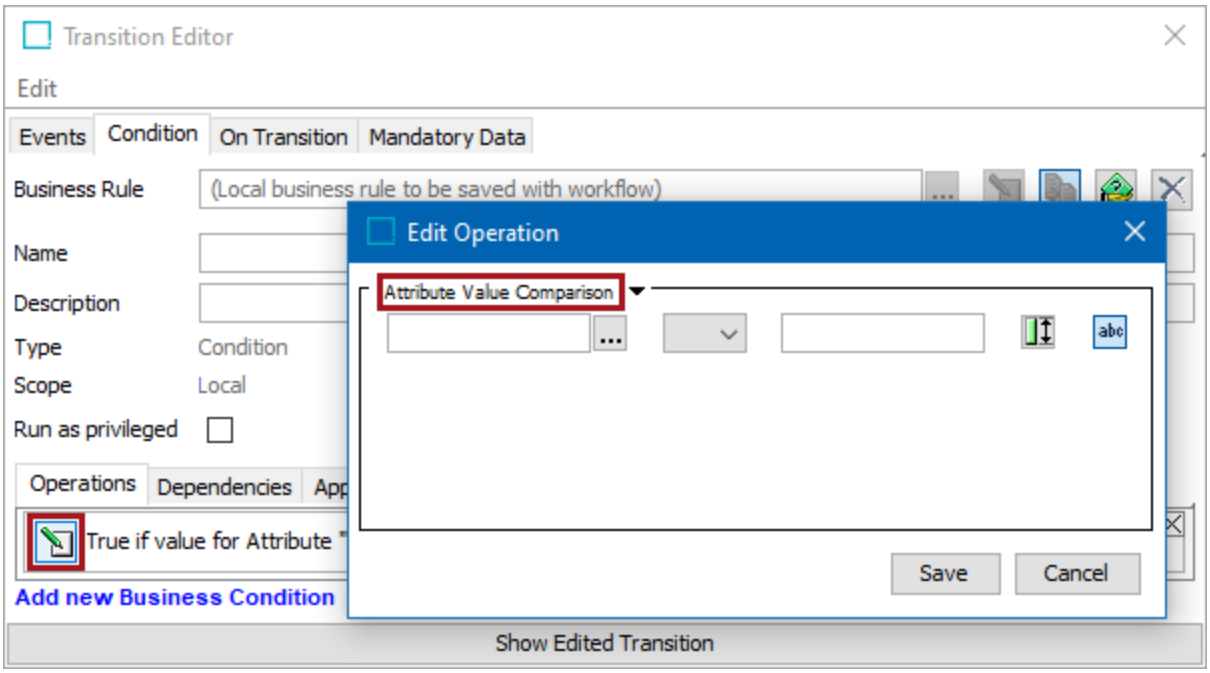
Data validation can be carried out using any number of functions and/or JavaScript. Additional information on more advanced topics can be found in the **Business Rules** section of the documentation. The below provides a simple example of data validation using the attribute value comparison function.

The below example will check to ensure that an attribute, List Price, has a value greater than or equal to .99. It assumes that a workflow has already been created and a transition exists on which the condition can be added.

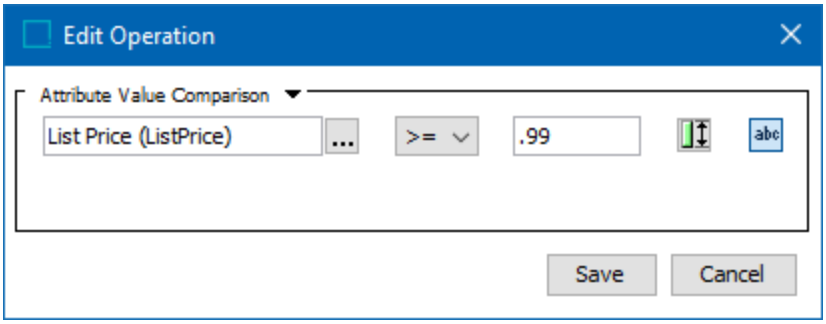
1. Double-click on the transition, select the **Condition** tab, then click the **Add new Business Condition** link.



2. Click the Edit Operation icon, select Attribute Value Comparison from the dropdown.

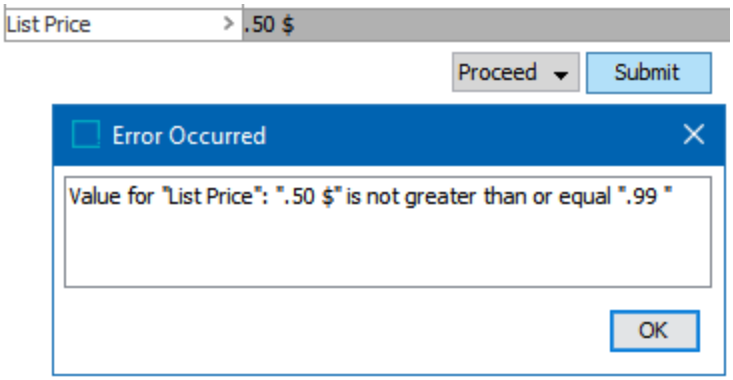


3. Click the ellipsis button (...) to select an attribute. Choose an operator from the dropdown, insert a value for comparison, and click **Save**.

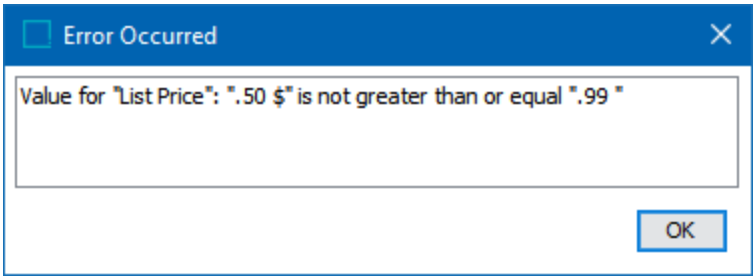


4. Click the [X] in the upper right corner of the Transition Editor to close it, then save and exit the workflow.

Users attempting to submit a task that does not meet the condition will receive an error.



Note that by default, the condition also checks for the presence of a value, returning false when no value is present.



Workflow Variables

When working with Workflows there are two different types of variables available of which one comes in two variants:

JavaScript Variables - Normal JavaScript variables can be declared in business actions and conditions. The scope of these variables is limited to the action / condition where they are declared and they are only persisted during the execution of the action / condition.

Workflow Variables - Workflow variables are string variables defined in the Workflow Designer. The variables are global to the workflow instance and the values are stored on the object-in-workflow relation and thus persisted as long as this relation remains. Workflow variables come in two flavors, 'bound' and 'unbound'.

Bound Workflow Variables

A bound workflow variable is bound to a STEP Attribute and gets its validation from the STEP Attribute. Apart from getting the validation from a STEP Attribute, bound workflow variables also get the corresponding attribute editors when a user chooses to display them in workbench views (see the **Views and Mappings for Workflows** section of the **Workflows** documentation).

Note: The attributes to bind to must be externally maintained, non-dimension dependent description attributes and should not be valid for any object types.

Unbound Workflow Variables

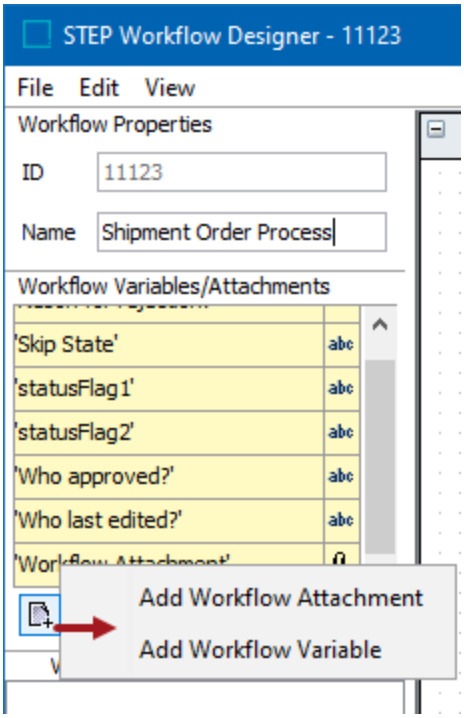
When a workflow variable is not bound to an attribute, it is a simple string variable with a default workbench view editor.

The value of unbound variables can be the result of an expression. In the expression panel, a reserved variable named 'value' is used. Below a script / expression is shown that will set the value of the variable to be the ID of the current node. Expressions are re-calculated each time the variable is used.

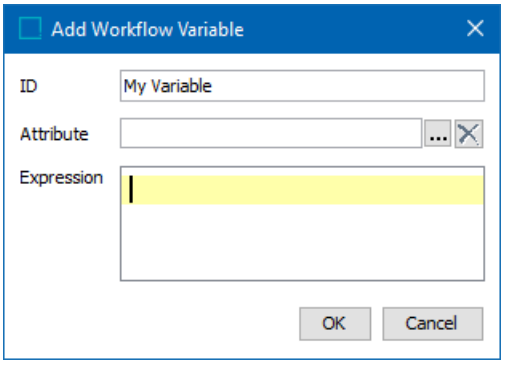
```
value = node.getID();
```

Creating a Workflow Variable

To create a workflow variable press the add icon and select Add Workflow Variable in the 'Workflow Variables/Attachments' section in the left side of the Designer.



This will bring up a dialog where the user can enter the ID of the variable and optionally select to bind it to an 'Attribute' or add an 'Expression'.



As noted in the **Business Rules in Workflows** section in the **Workflows** documentation, workflow variables are manipulated through the WorkflowInstance interface. Thus the value of a workflow variable can be obtained from a business action as follows:

```
var valueStoredInVariable = node.getWorkflowInstance
(workflow).getSimpleVariable("Variable ID");
```

And likewise a simple value for a workflow variable could be set with:

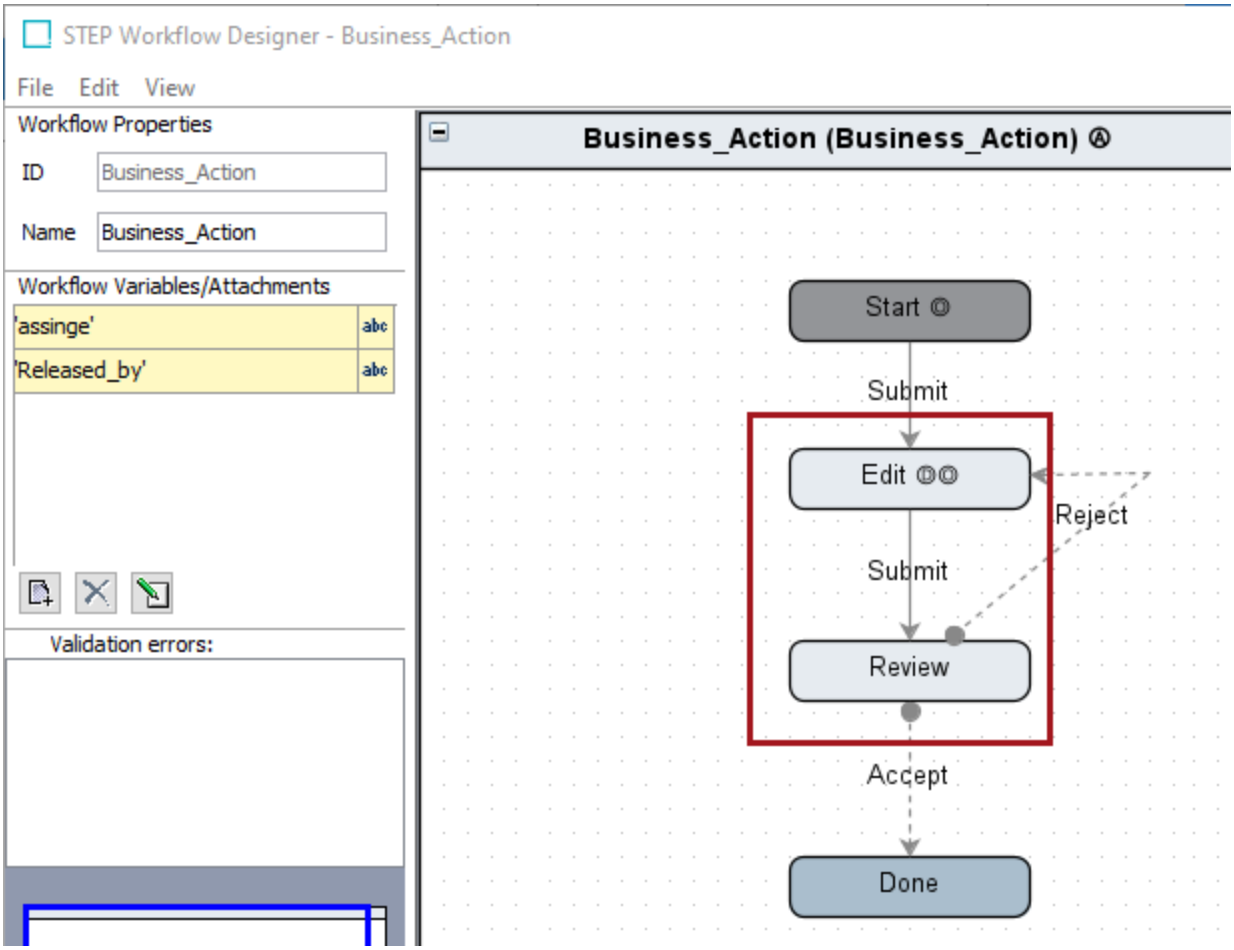
```
node.getWorkflowInstance(workflow).setSimpleVariable("Variable ID",
"Value to be set");
```

While the above applies to both bound and unbound variables, for bound variables a value object can be obtained. This can be done with:

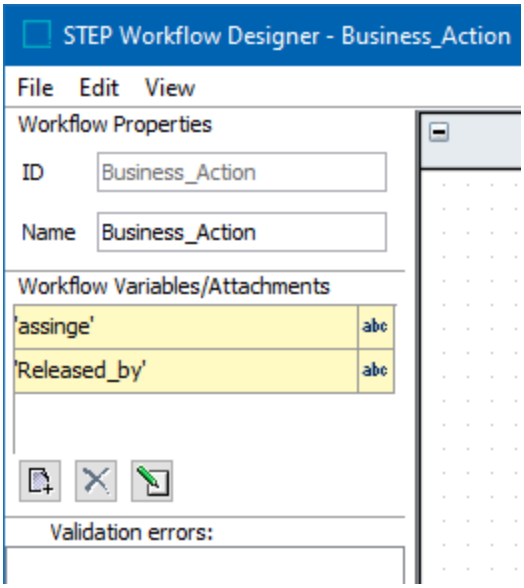
```
var valueObj = node.getWorkflowInstance(workflow).getValue("ID of
Attribute the variable is bound to");
```

Workflow Variables in Use

Consider a user group which consists of five users. All of these five users have access to a particular workflow, and can submit a product from the Edit to the Review state. For this workflow, because there is a requirement to identify which of these five users submitted a particular product, a business rule has been set up to save the details to an attribute. This business action is set on the desired state in the State Editor on the 'On Exit' tab. In this case it is on the Edit sate.



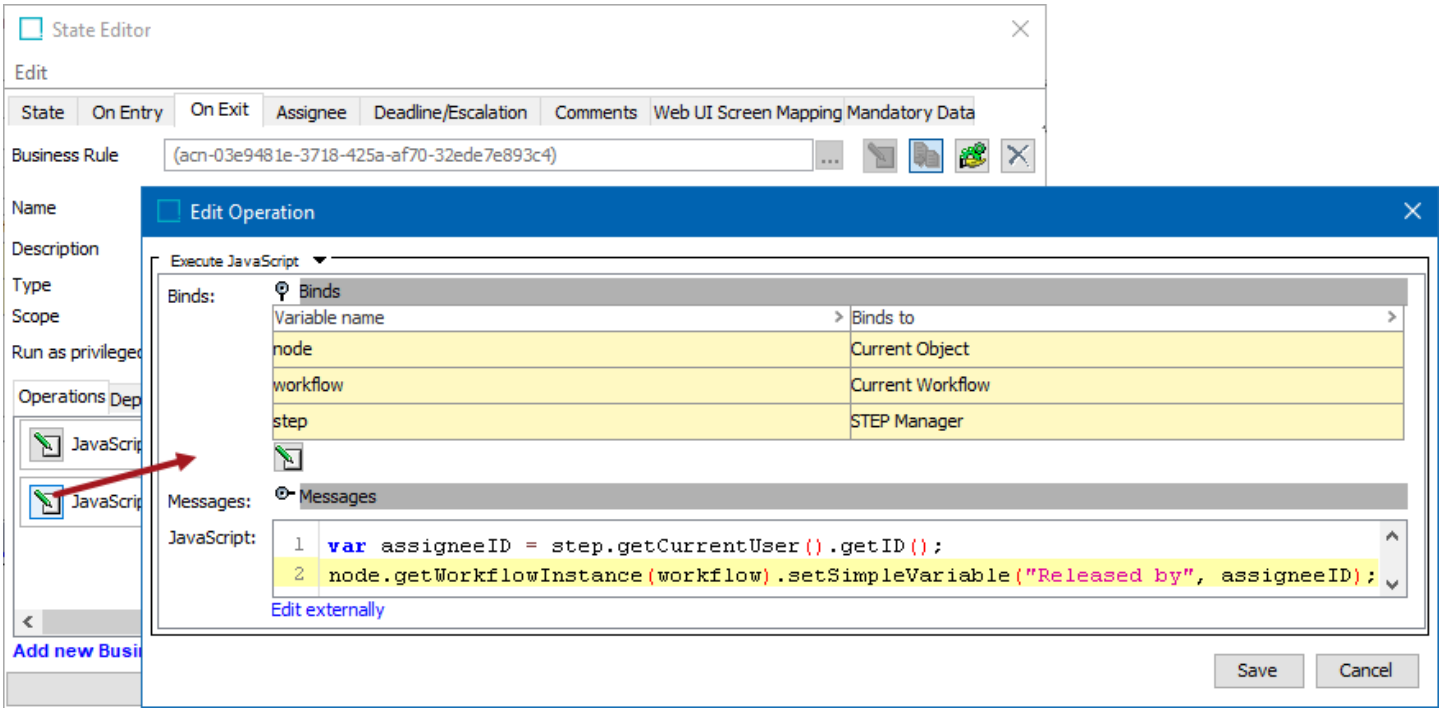
To create this business rule, a workflow variable first needs to be defined, which is 'Released by' for this workflow.



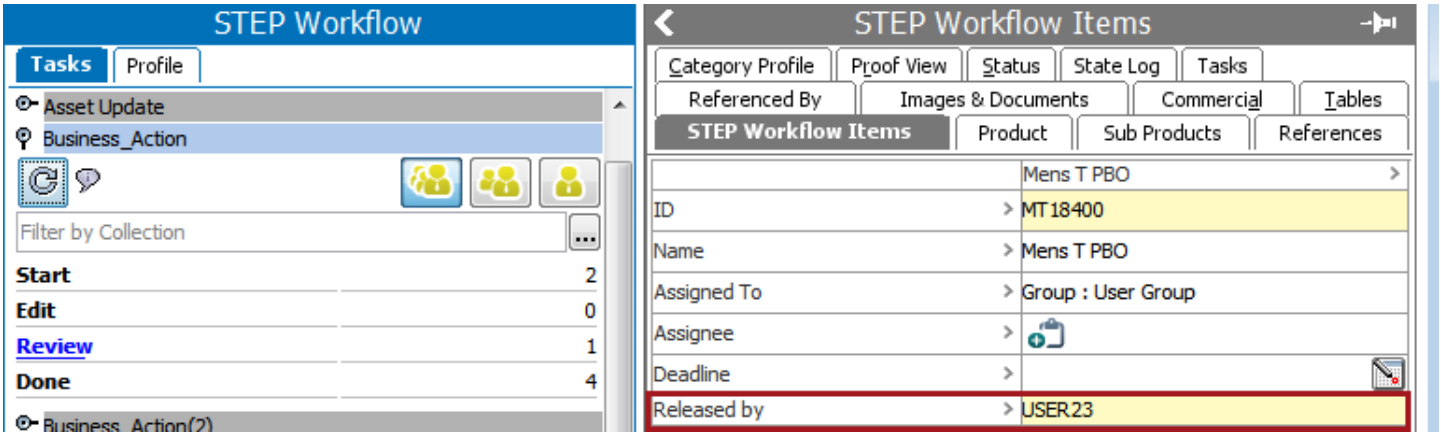
Then a business rule needs to be written to call the workflow variable as mentioned below:

```
var assigneeID = step.getCurrentUser().getID();
node.getWorkflowInstance(workflow).setSimpleVariable("Variable ID",
assigneeID);
```

Note: In this formula, "Variable ID" should be replaced with the ID of the actual workflow variable for the workflow in use.



With this business action in place, when a product is submitted, the business rule is executed and the user ID of the user who submitted the product is stored. This is visible in the STEP Workflow Items tab if this was mapped in the views and mappings View Editor.



For more on setting up views and mappings, see the **Views and Mapping for Workflows** topic in the **Workflows** documentation.

Workflow Variables in Web UI

Workflow variables are also supported in Web UI, where they can be added to relevant components using the Workflow Variable or Workflow Variable Header components. In addition, workflow variables can be added to submit actions (Submit Action or Submit from Grid Action).

Web Services for SOAP Requests for Workflows

Complete documentation for Web Services functionality related to workflows can be found in the STEP API documentation at [system]/sdk or by clicking the **STEP API Documentation** button on the STEP Start Page. This topic provides some basic information and an example of a simple SOAP request, but should not be considered comprehensive.

The STEP Web Services API contains the following methods related to STEP Workflows:

startWorkflow - Method for starting a Workflow on an Object

triggerWorkflowEvent - Method for triggering a Workflow Event for an Object in a State in a Workflow

deleteWorkflowInstance - Method for removing an Object from a Workflow. I.e., deleting the instance.

getTasks - Method for retrieving info about which States an Object is in in specified Workflow. For each State apart from the ID, a STEP url for the Assignee and the time of entering the State will be returned.

initiateItemInStateflow - Legacy version of startWorkflow

triggerStateflowEvent - Legacy version of triggerWorkflowEvent

removeItemFromStateflow - Legacy version of deleteWorkflowInstance

getItemStateInfo - Legacy version of getTasks

An example using startWorkflow is below:

```
<soapenv:Envelope
xmlns:soapeny="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns="http://stibo.com/step/ws/step/1.0"
xmlns:ns1="http://stibo.com/step/ws/types/1.0">
   <soapenv:Header/>
   <soapenv:Body>
      <ns:startWorkflowRequest>
         <ns:accessContext>
            <ns1:userName>[Username]</ns1:userName>
            <ns1:password>[Password]</ns1:password>
            <!--Optional:-->
            <ns1:passwordType></ns1:passwordType>
            <!--Optional:-->
            <ns1:contextUrl>step://context?id=[Context
ID]</ns1:contextUrl>
            <!--Optional:-->
```

```
            <ns1:workspaceUrl>step://workspace?id=[Workspace
ID]</ns1:workspaceUrl>
         </ns:accessContext>
         <ns:nodeURL>step://product?[Node ID]</ns:nodeURL>
         <ns:WorkflowID>[Workflow ID]</ns:WorkflowID>
         <!--Optional:-->
         <ns:message>[Message]</ns:message>
      </ns:startWorkflowRequest>
   </soapenv:Body>
</soapenv:Envelope>
```

Workflow Attachments

While not the only reason, 'Workflow Attachments' are typically used for holding files that are necessary for users handling tasks in states. This could be written instructions, images, data sheets etc. Once the attachment is in place, when a user receives or gets a specific task(s), they can access the attachment and follow the specified instructions / guidelines.

Using workflow attachments a user can attach files to workflow instances. Also, like the workflow variables, workflow attachments are global to the workflow instance.

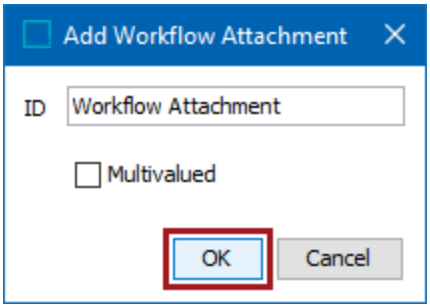
Note: Users should be familiar with the navigation to the STEP Workflow Designer and defining the views and mappings. See **Creating a Workflow** in the **Workflows** documentation for more on the workflow designer. Also, see **Views and Mappings for Workflows** in the **Workflows** documentation for additional information on view and mapping.

Workflow Attachment Configurations

As described in the **Configuring Views for Workflows** section in the **Workflows** documentation, the 'WorkflowAttachment' component allows Users to add, open, download and delete workflow attachments through views.

Attachments are defined in the left pane of the Workflow Designer. To add a workflow attachment:

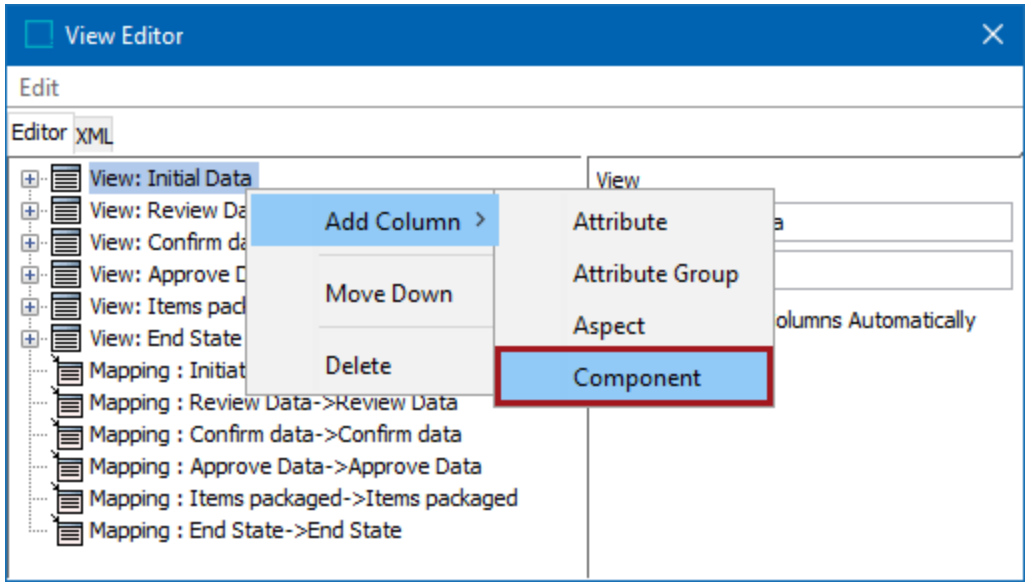
1. Click on the 'Add Workflow Variables/Attachments' icon,  , and select the option **Add Workflow Attachment**.
2. In the dialog that appears, enter the ID and click **OK**.



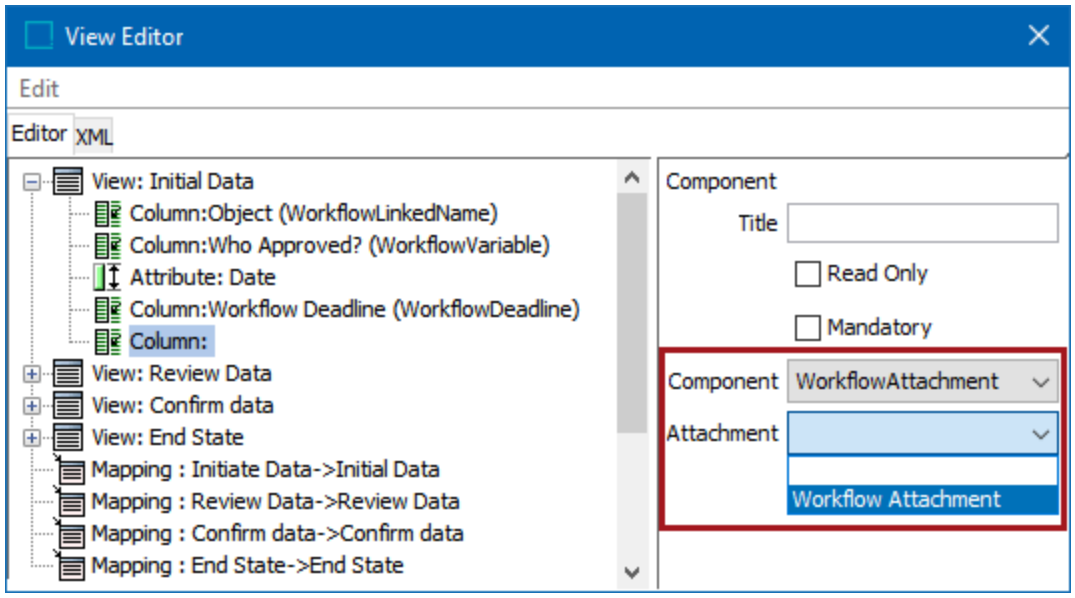
If multiple attachments are required then check the option for 'Multivalued'.

Note: If a user needs to edit or delete a workflow attachment added in this section, they can click on the edit icon,  , or they can click on the remove icon,  , to make any necessary changes.

3. Next go to Edit > **Edit Workbench Views and Mappings**.
4. In the View Editor, right-click the view that needs an attachment added to it, and select Add Column > **Component**.



5. Under the Component dropdown, select 'Workflow Attachment'. Once selected a dropdown to add an 'Attachment' will appear. Select the attachment that was just created in step 2. For this example it is called 'Workflow Attachment.'



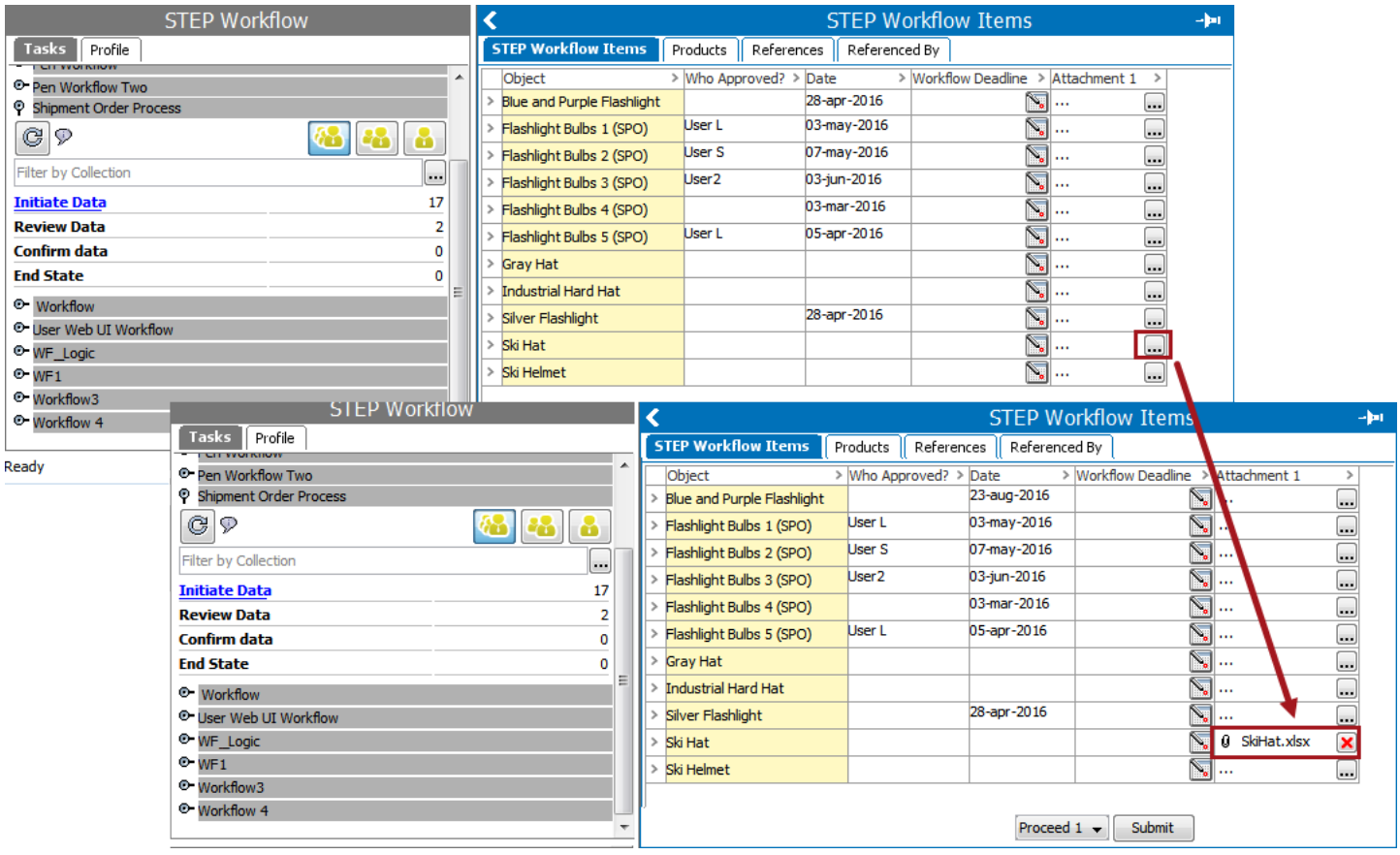
6. Enter a title and then decide if either 'Read Only', or 'Mandatory' need to be checked before exiting the View Editor and hitting 'Save and Exit' in the STEP Workflow Designer. Note that neither one of these has to be checked. If 'Read Only' is left unchecked then a user can download the attachment and upload files. These files will be attached to the workflow instance.

Component	Returned Value
Read Only	If marked as this the add and delete options will be disabled, but users will still be able to open and download attachments. However, any uploaded files will not be stored in the STEP database. Instead, they will live in the application server file system in a path specified via the configuration property 'Install.STEPWorkflowArea'.
Mandatory	Tell the user that the attachment is required information.

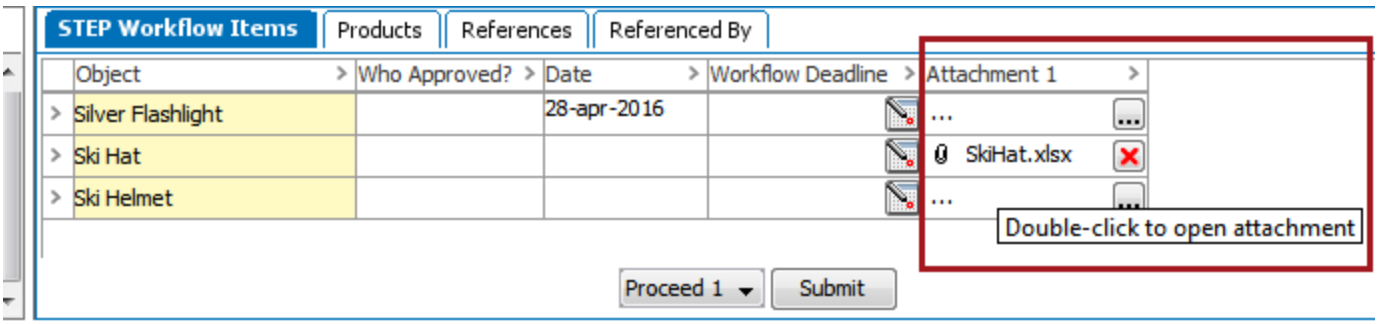
Attaching a File in a Workflow

Now that the configurations are in place, the steps below show a user how to attach and remove a file.

1. Navigate to the STEP Workflow tab, select the desired workflow, and click on the corresponding state that the workflow attachment was mapped to in the above steps.
2. In the STEP Workflow Items tab, click the ellipsis button (...) next to the desired product and select the file from the system that is to be attached.



3. Double click on the attachment to open it.



4. To remove the attachment, click on the red 'x' button, .

It is worth noting that workflow attachments are instance specific, and values will be deleted together with the instance.

In the public API you can obtain a 'WorkflowAttachment' object through the 'WorkflowInstance' interface via 'WorkflowInstance.getAttachmentByID'("Attachment ID") and have various methods available on the returned object.

Note: The files referenced by attachments are stored on the application server in the directory pointed to by config property "Install.STEPWorkflowArea". If you are on a clustered environment, the directory should be on a shared drive. Workflow attachments are not stored as assets in STEP.