Adobe Workfront

ADOBE WORKFRONT GUIDE

# Fusion Cookbook

Real scenarios from Adobe Workfront customers and consultants

Adobe Workfront

Adobe

# Table of Contents

*Please note that the scenarios showcased in this Cookbook only represent a small portion of Adobe Workfront Fusion's capabilities. Please stay tuned for more Adobe-to-Adobe scenarios in a follow-up Adobe Workfront Fusion Cookbook!*

Adobe Workfront

FUSION REPORT RECIPES

# Getting Started with Adobe Workfront Fusion Templates

**Prep:** 30 minutes

**Yield:** Select a template from the template library. Follow guided instructions to build a scenario from the template. Consider further fine-tuning to meet your needs

**Benefit:** Starting from an Adobe Workfront Fusion Template reduces development cycles and allows you to see what is possible without a heavy lift

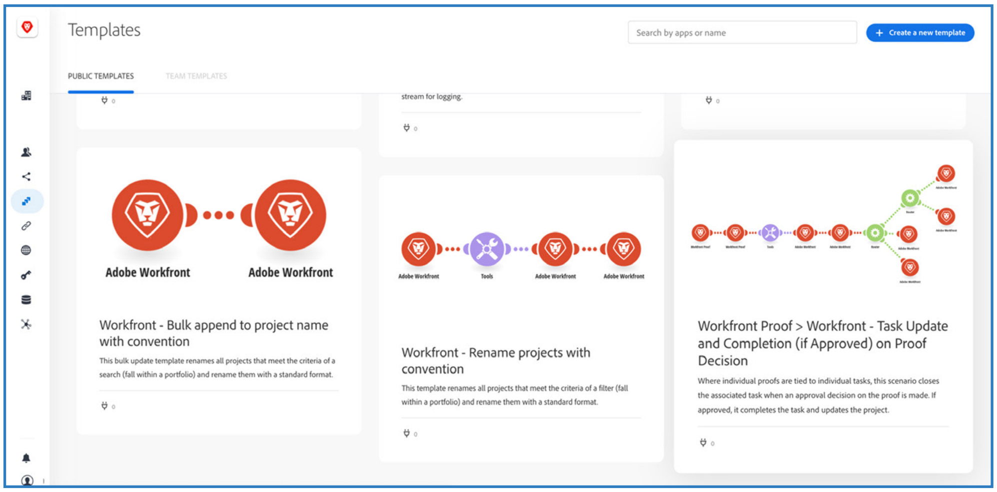
**Serves:** Process Owners, Integration Architects/Developers, System Admin

**By:** Ewan Hruska

# Getting Started with Adobe Workfront Fusion Templates

## Detailed instructions

**Step 1:** After logging in, you can access Fusion Templates by selecting the **Templates** icon on the left sidebar menu (circled below).




**Step 2:** Once you find a template that meets your needs from the **Template** library, select it by clicking on **Create a new scenario from template** from the template overview window. Follow the unique prompts for each template in order to build it according to your needs.

Adobe Workfront

FUSION REPORT RECIPES

# Convert Approved Issue to Project

**Prep:** 90 minutes

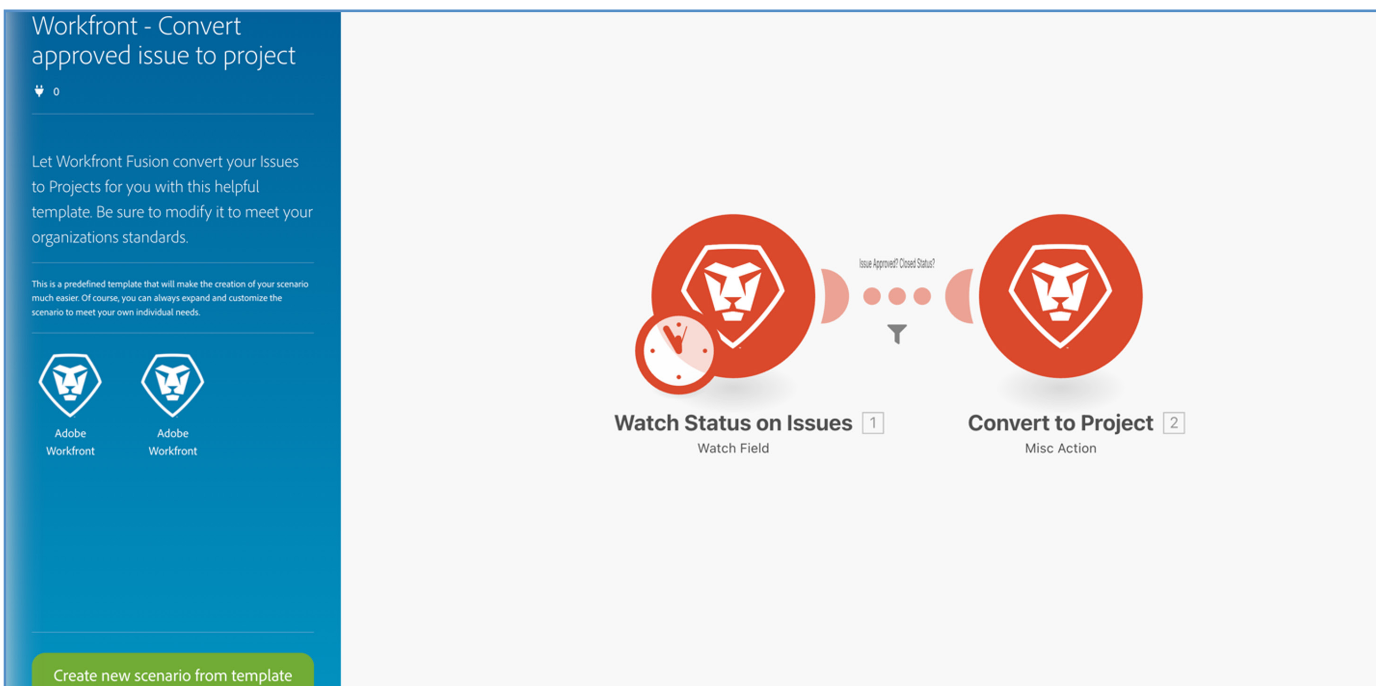
**Yield:** Convert Issues (Requests) to Projects based on specific criteria

**Benefit:** This template and the resulting scenario is based off an extremely common need to be able to convert approved issues to projects

**Serves:** Project Owners, Process Owners

**Modules:** Adobe Workfront

**By:** Ewan Hruska

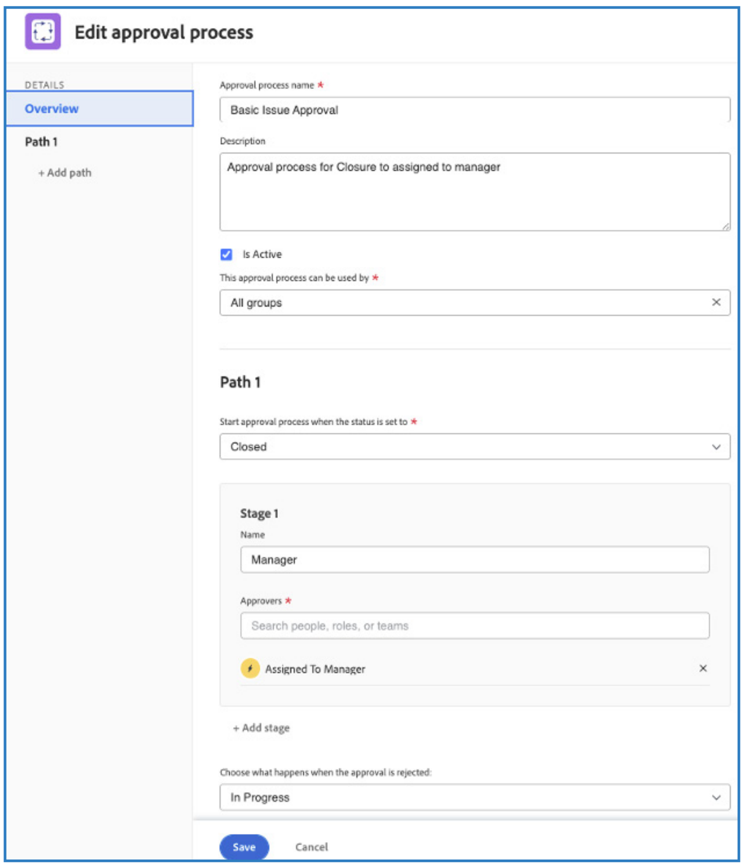


# Convert Approved Issue to Project

## Detailed instructions

**Step 1:** You need to have an [approval workflow setup] in your Workfront environment based on Closed status if you are running this as designed on Approved issues. Other statuses could be used for the approval workflow but a few small additional changes in the scenario would be needed.

**Step 2:** From inside Workfront go to the **Main Menu** > **Setup** select **Processes** > **Approvals** and go to the Tab. Create a Basic Approval Workflow that looks something like the following one. The key point for the template to work as is, are to **Start approval process when the status is set to: Closed.**




**Step 3:** To kick off the approval process when an issue is closed, you need to attach the Approval Process we just created with the issue. This can be done in an individual issue's settings (Good) OR within the project for any issue created (Somewhat Scalable) OR in a project template (Enterprise Scalable).

# Convert Approved Issue to Project

## Detailed instructions

**Step 4:** The high-level workflow overview of the template is:


**Pro-Tip:** There are a ton of different things to map when converting issues to projects. Below you will find a subset of some of the settings you could change in your scenario:

1. **Project Name:** This example changes the project name to include the word "PROJECT" prior to the existing name. This could be easily changed in the name area from the Convert To Project module options. Check out the Project Rename Templates for some ideas on Project Name structure.

2. **Templates:** You may (probably) want the new project to use a specific project template…if so you can add the project template through the Template ID field. Pull the template id from the URI of the template in Workfront.

3. **Conversion Options:** Remove preserveIssue if you do not want to maintain the original Issue. We recommend keeping it for tracking purposes in most situations.

Standard Options:

a. preserveIssue

b. preservePrimaryContact

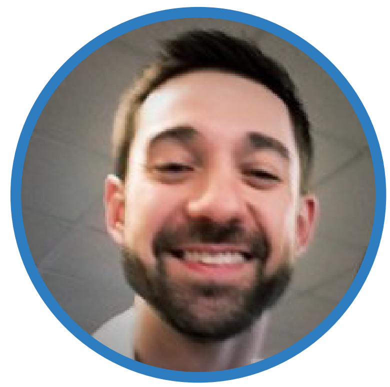
c. preserveCompletionDate

Adobe Workfront

Adobe

FUSION RECIPES

# Move New Documents from Original Issue to Resolving Project

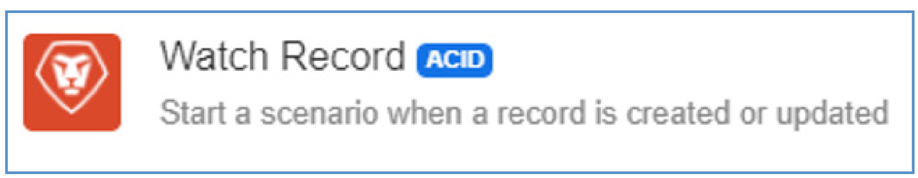


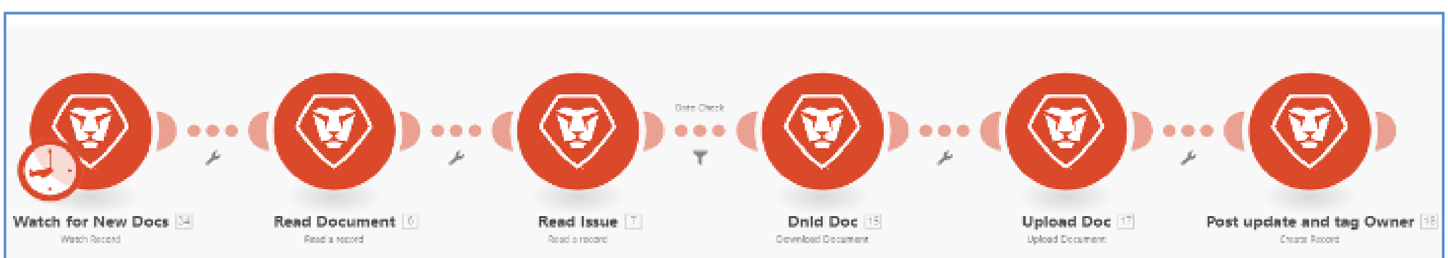
**Prep:** 10 minutes

**Yield:** Avoid missing new documents added to the original issue

**Serves:** Project Owners and Requesters

**By**: Brandon Pritchard (Nationwide)



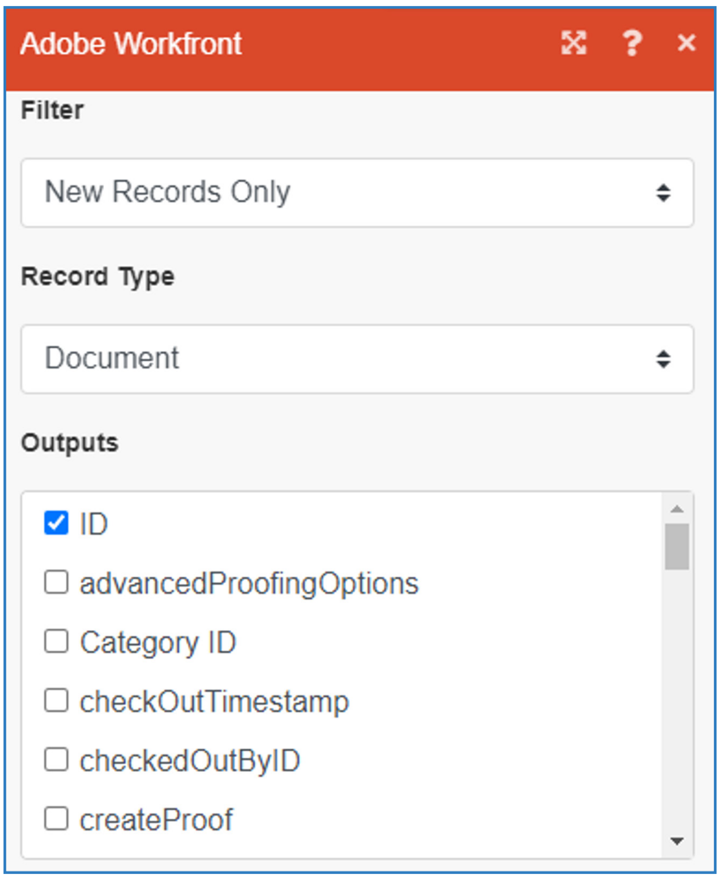



# Move New Documents from Original Issue to Resolving Project

## Detailed instructions

**Step 1:** Use the **Watch Record** Module to look for new Documents.

- Filter: New Records Only
- Record Type: Document
- Record Origin: New and Updated

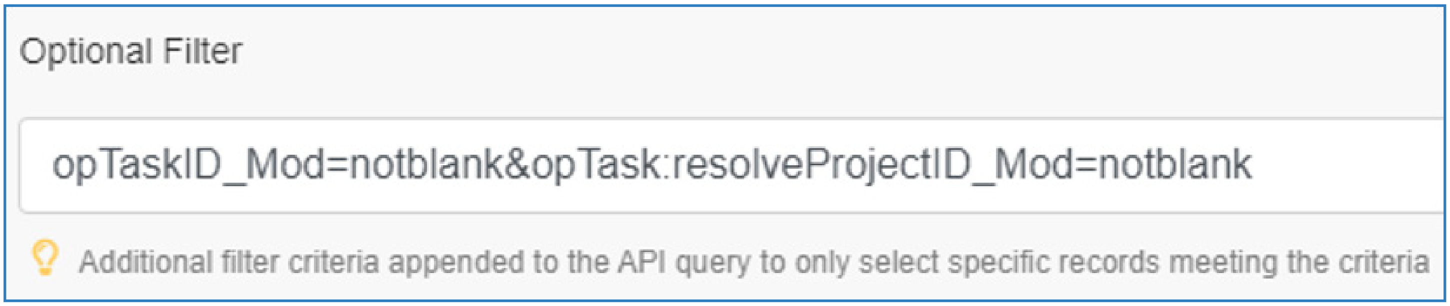


# Move New Documents from Original Issue to Resolving Project

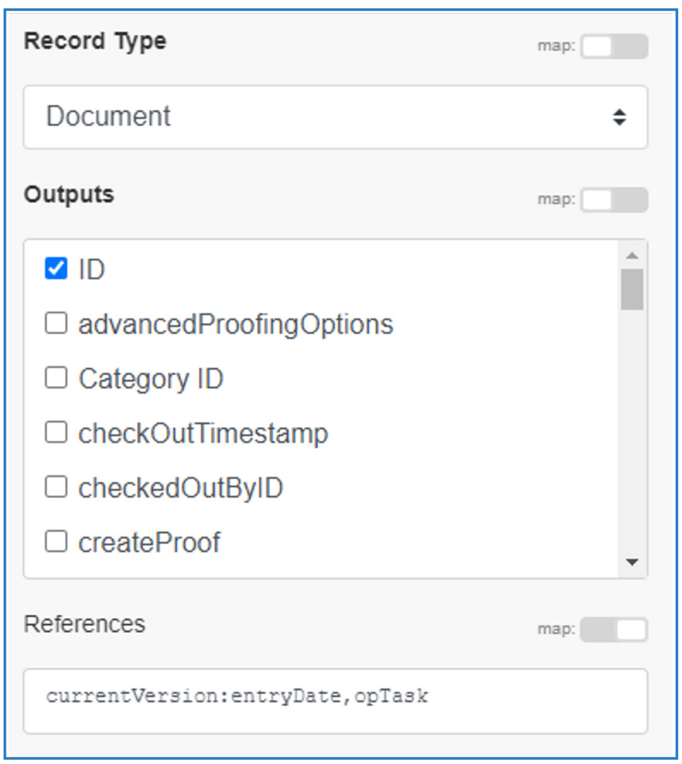
## Detailed instructions

**Pro Tip:** You can use the "optional filter" field to add additional filters. The syntax is the same as a text mode filter. For multiple filters, use the "&" as an AND function.

In the filter example below, we are looking for documents that have an issue ID and a resolving project ID.

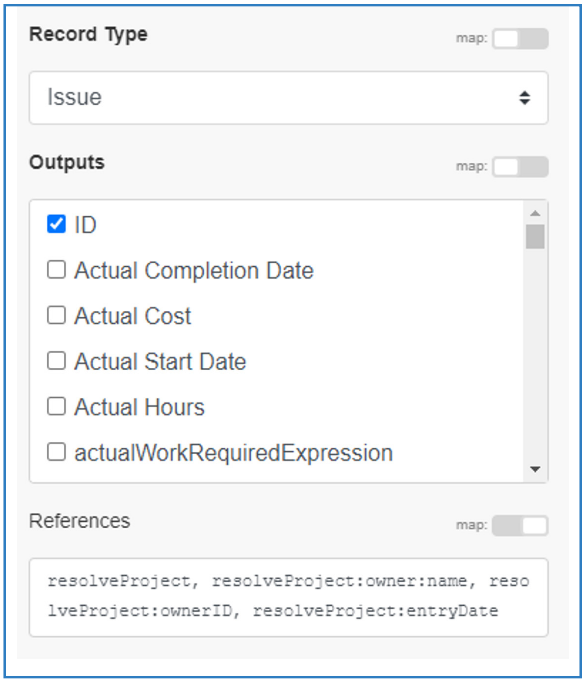



**Step 2:** Add a **Read a Record** Module to read the document data.



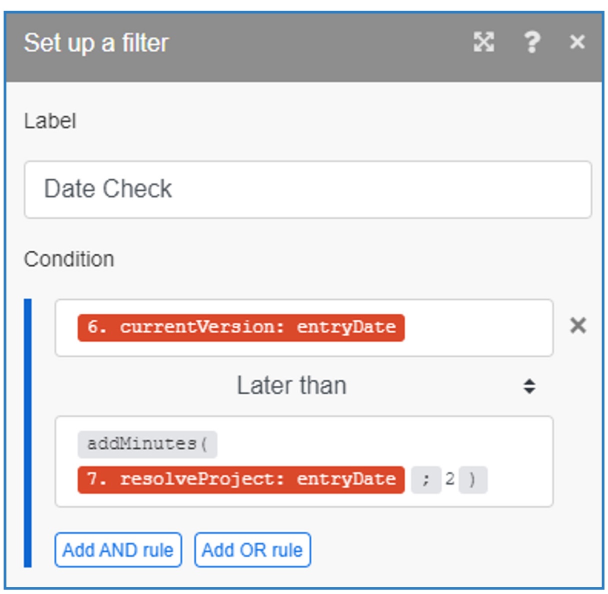
# Move New Documents from Original Issue to Resolving Project

**Step 3:** Add a **Read a Record** Module to read the document data.



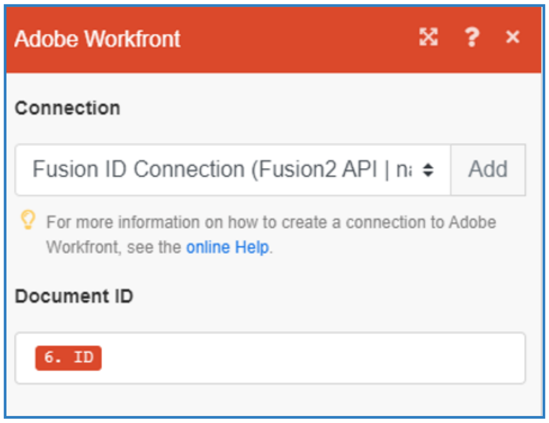

**Pro Tip:** Use the **References** section to jump data tables and pull referenced data.

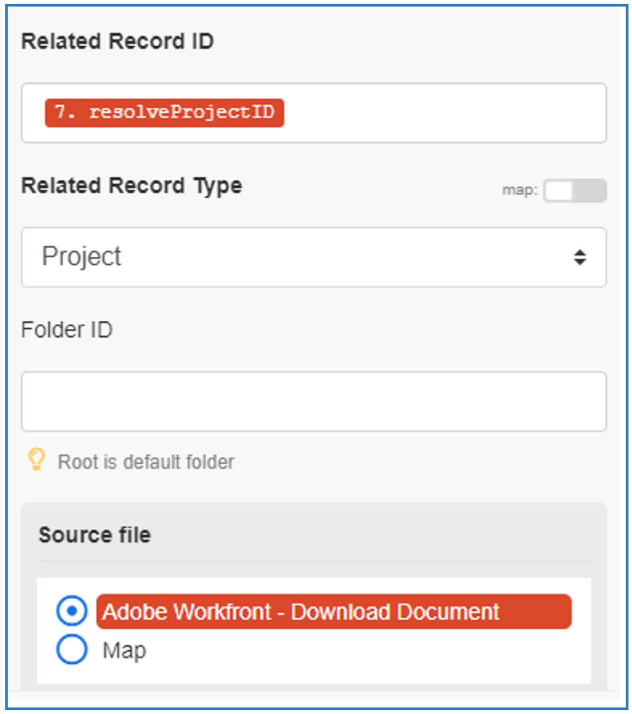
**Step 4:** Use a filter to only allow through new documents added to the issue **after** it was converted into a project. Adding on additional minutes can help ensure the entry dates do not match.



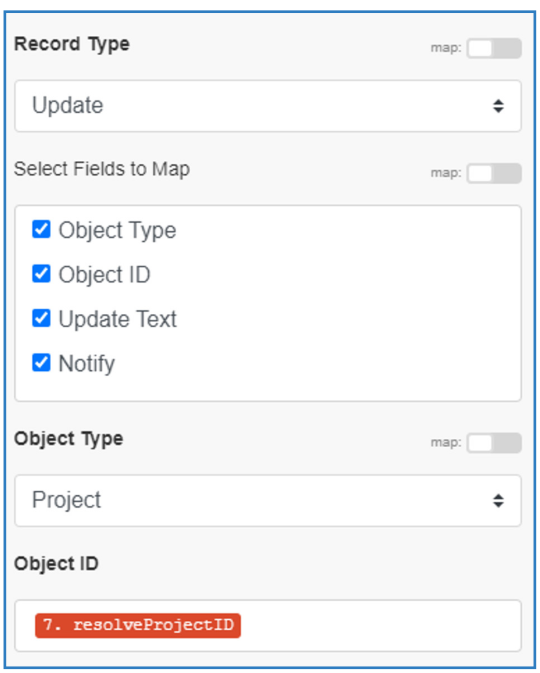
# Move New Documents from Original Issue to Resolving Project

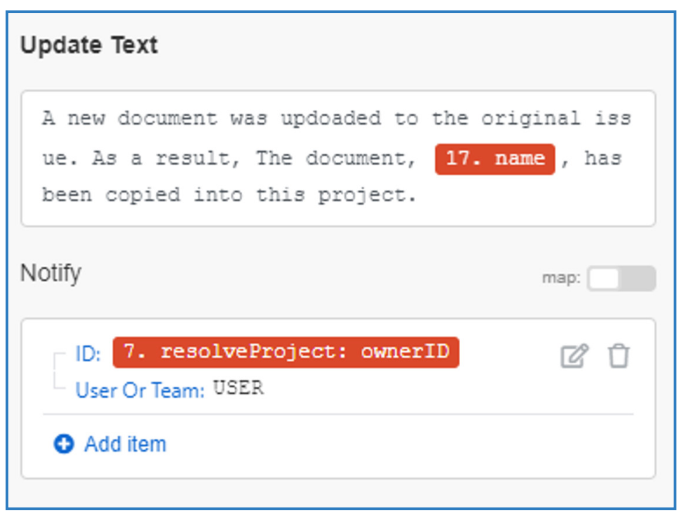
**Step 5:** Add a **Download Document** Module and an **Upload Document** Module to download the new document and upload it to the resolving project.




**Step 6:** Add a **Create Record Module** to post an update to the resolving project and notify the project owner.

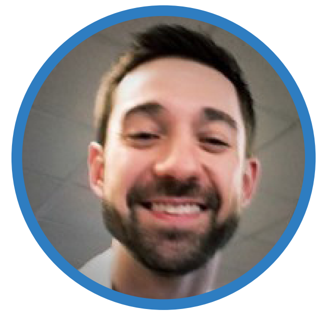


Adobe Workfront

FUSION REPORT RECIPES

# Proof Settings Automation with Custom Form Update



**Prep:** 15 minutes

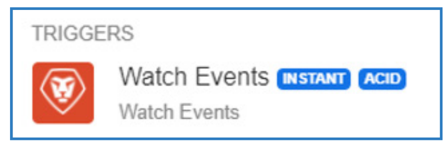
**Yield:** Create a Proof Custom Form in the Account Settings area of Workfront Proof

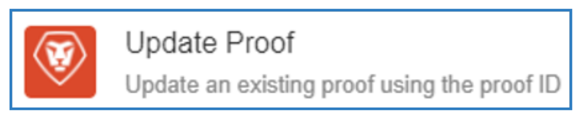
**Benefit:** Reduce the number of clicks, set standardized settings, and enable project association

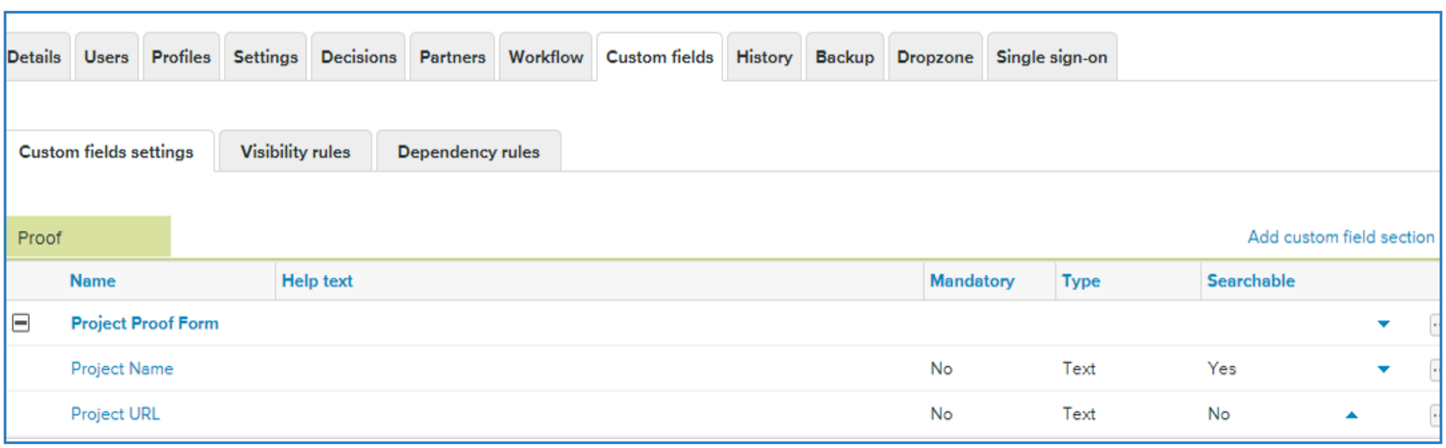
**Serves:** Project Owners, Proof Owners

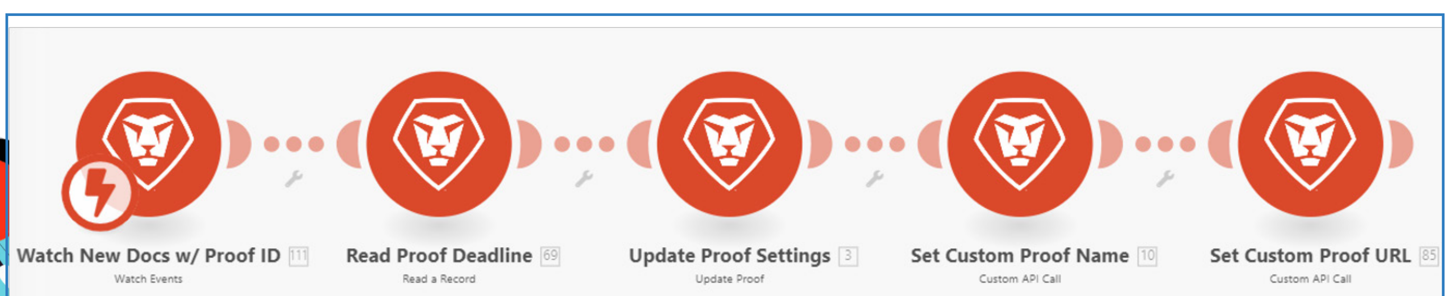
**Modules:** Adobe Workfront, Adobe Workfront Proof

**By:** Brandon Pritchard (Nationwide)



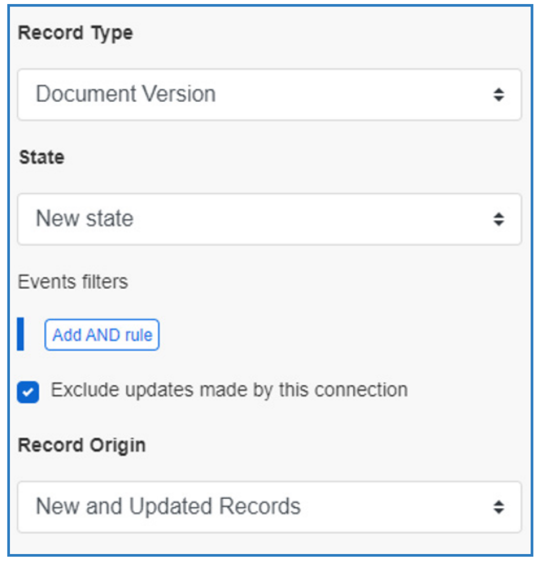



# Proof Settings Automation with Custom Form Update

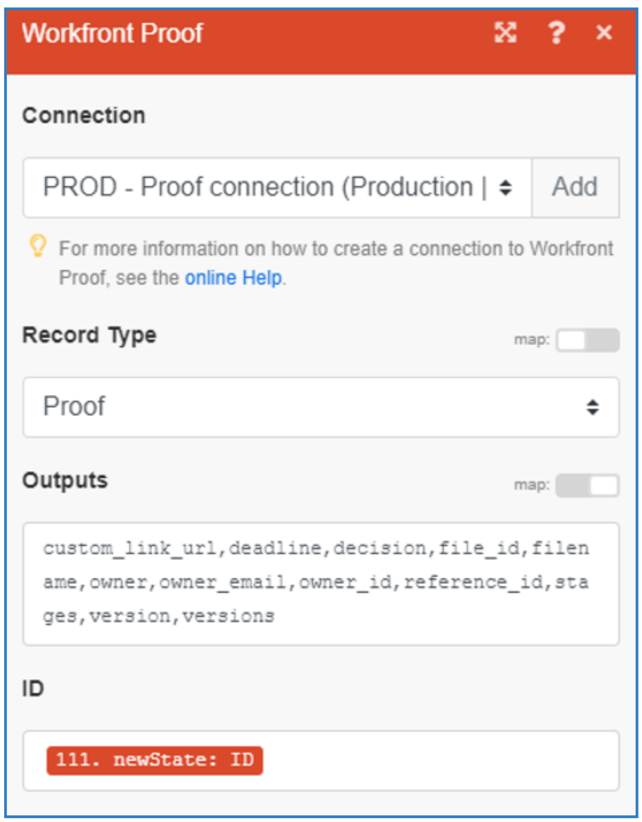
## Detailed instructions

**Step 1:** Use the **Watch Events** Module to look for Document Version s.s

- Record Type: Document Version
- State: New State
- Record Origin: New and Updated




**Step 2:** Add a **Read a Record** Module to read the Proof information

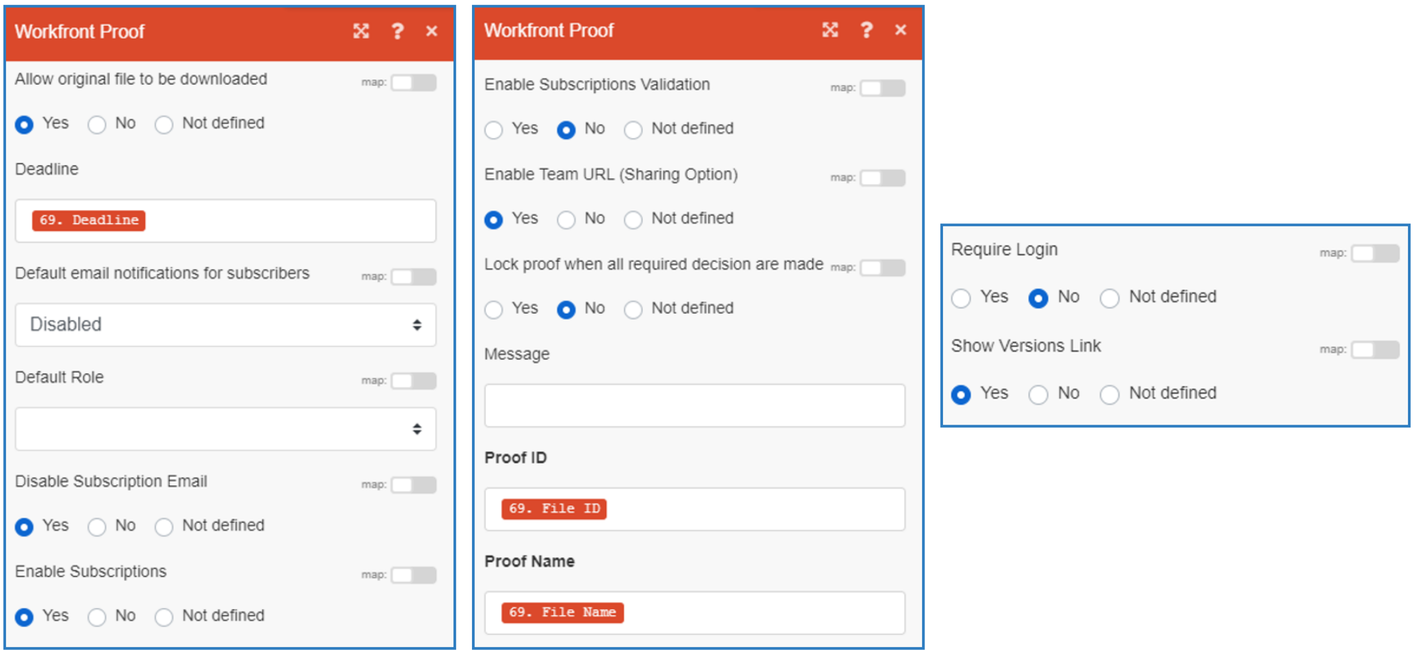



**Pro Tip:** You can type the desired fields into the output section by toggling the "map" button on.

# Proof Settings Automation with Custom Form Update

## Detailed instructions

**Step 3:** Add an **Update Proof** Module to default the proof settings to your specifications.




**Step 4:** Add a **Custom API Call** Module to set the project Name on the custom form.

- Method: setProofCustomData
- Use the Body (XML) section to identify the File ID or Proof ID and set the custom value.

# Proof Settings Automation with Custom Form Update

## Detailed instructions

**Step 5:** Add a **Custom API Call** Module to set the Project URL on the custom form.

- Method: setProofCustomData
- Use the Body (XML) section to identify the File ID or Proof ID and set the custom value.

**Pro Tip:** You can type in the data from previous modules without mapping them in the previous module.

Syntax: "*{{module number.data field}}*"

**Method**

```
setProofCustomData
```

**Body (XML)**

```
<FileID> 3. File ID </FileID>
    <Data>
        <item>
            <name>Project Proof Form</name>
            <fields>   projectID
                <item>
                <name>Project URL</name>
                <type>1</type>
                <value>https://nationwide.my.wo
rkfront.com/project/ 3. File ID /documents</va
lue>
                <multiple>false</multiple>
                </item>
            </fields>
        </item>
    </Data>
```

Adobe Workfront

FUSION REPORT RECIPES

# Task Changes - Staffing

**Prep:** 1-2 hours per item

**Yield:** Leverage Fusion to apply project attributes based on related objects. This scenario covers 3 different use cases you can apply to your environment.

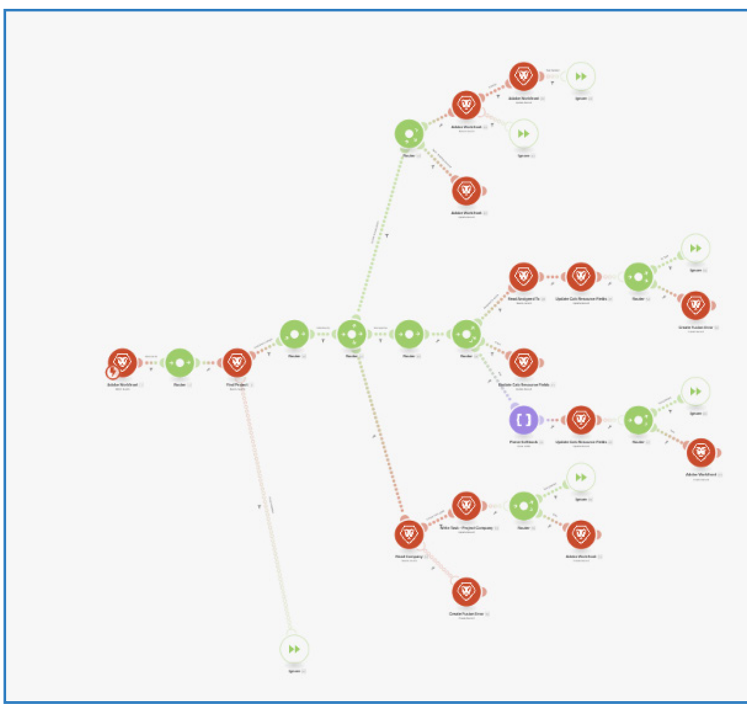
**Context:** We track staffing assignments and planned hours at the project level. We introduced "staffing tasks" that are at the top of each project plan. These tasks are the only tasks with planned hours, are tagged using custom fields, and pulled into custom reporting solutions we use to track capacity and utilization. This scenario automates maintenance, calculates data, and copies information to support Workfront's native task reports.

**Benefit:** Remove manual steps, summarize data across multiple projects, quarterly/ annual forecasting, cleaner data, more robust reporting, and more

**Serves:** Resource Managers, Report Builders

**Modules:** Adobe Workfront

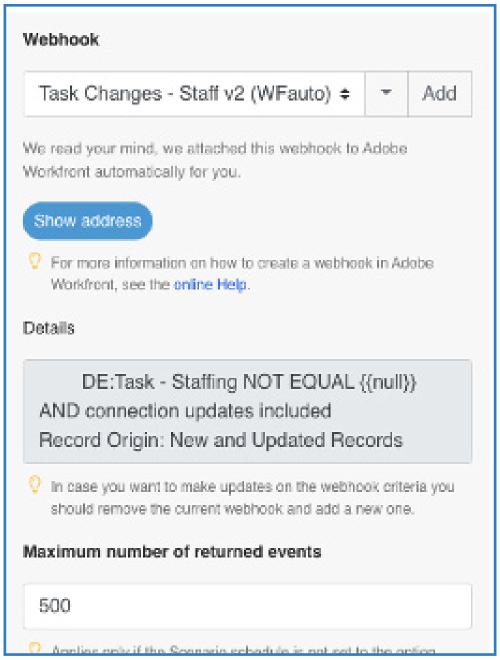
**By:** Jason Blunk



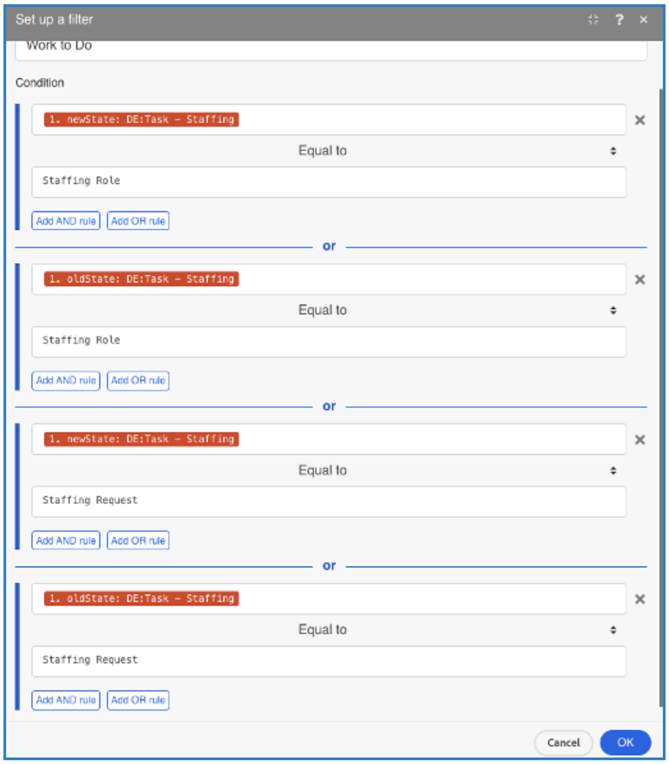
# Task Changes - Staffing

## Detailed instructions

**Step 1:** Create a webhook. In the detalis of your webhook, set parameters to filter down to only staffing tasks (as identified by a custom attribute on the task).



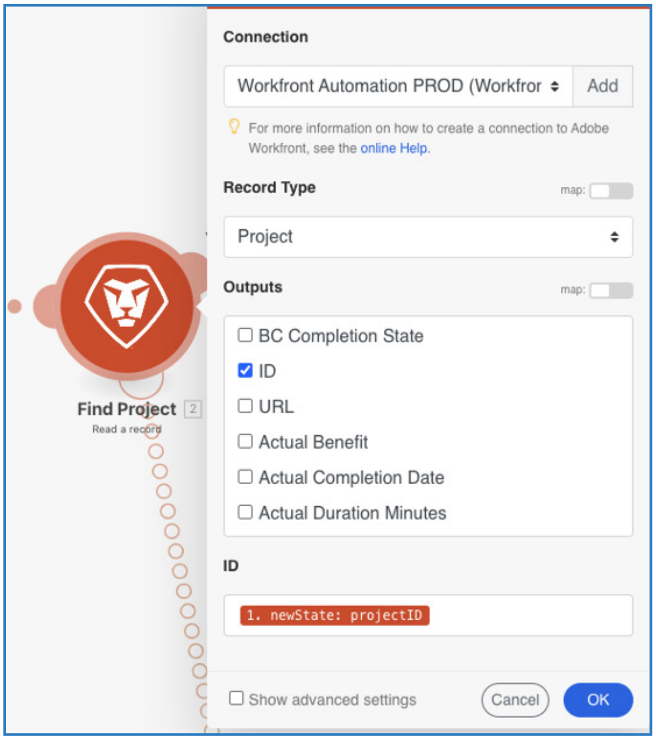

**Step 2:** Further narrow down your tasks using a filter. Set conditions according to your use case.

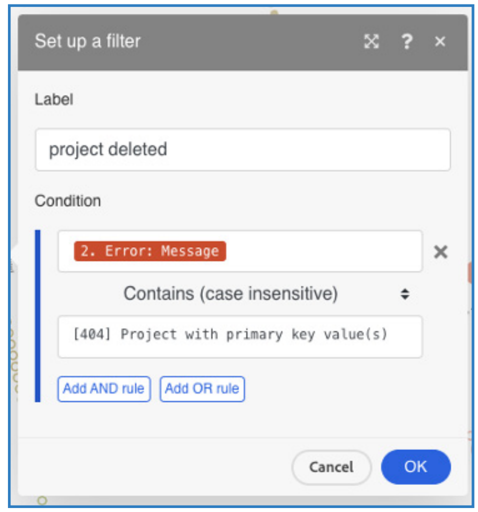


# Task Changes - Staffing

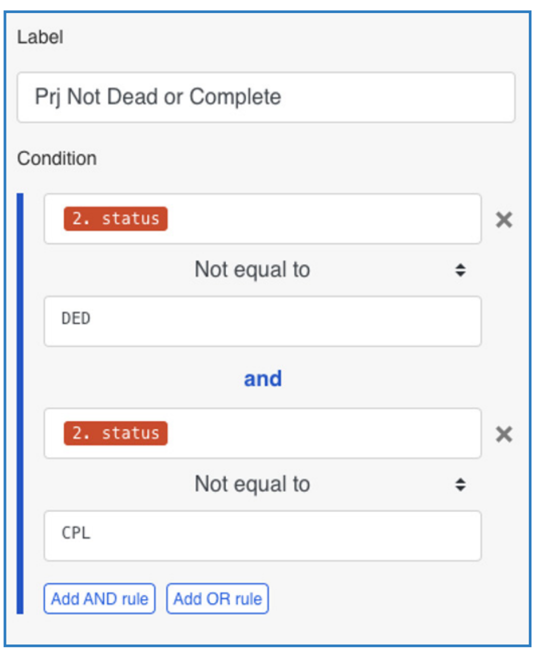
## Detailed instructions

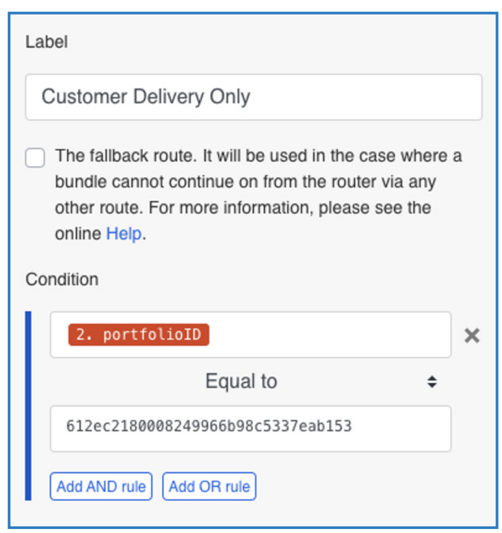
**Step 3:** Find related projects and get the project level data needed. You can also setup an Error Handler to ignore projects if they are deleted.




**Step 4:** Remove **Dead** and **Complete** projects because the system is unable to update it by itself. Set up filters for **Customer Delivery** portfolio only.

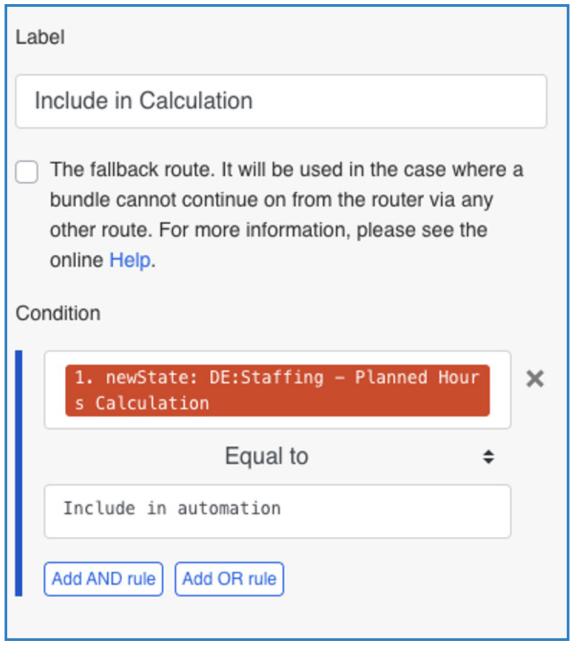


# Automating Data Attributes - Calculate Planned Hours

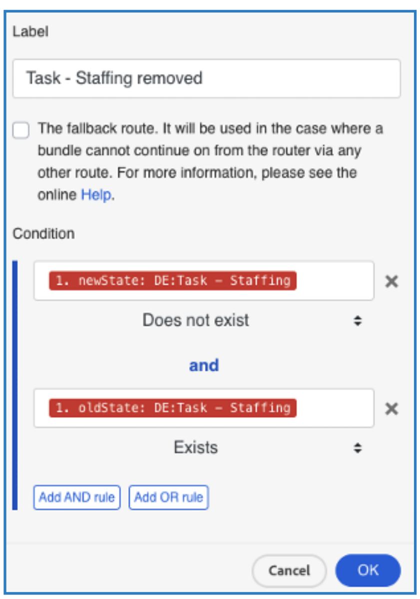
## Detailed instructions

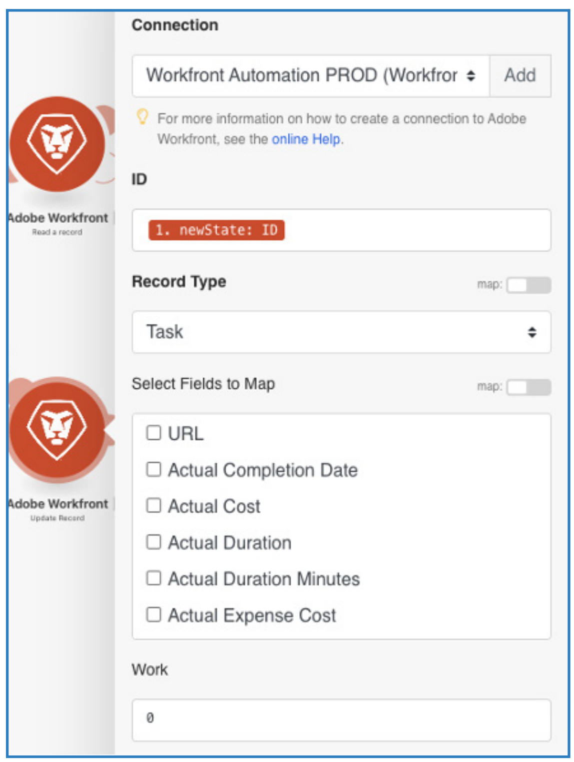
**Step 1:** Filter for only managed tasks.




**Step 2:** Filter for Staffing Removed and Update Work = 0.

- Work is "Planned Hours"

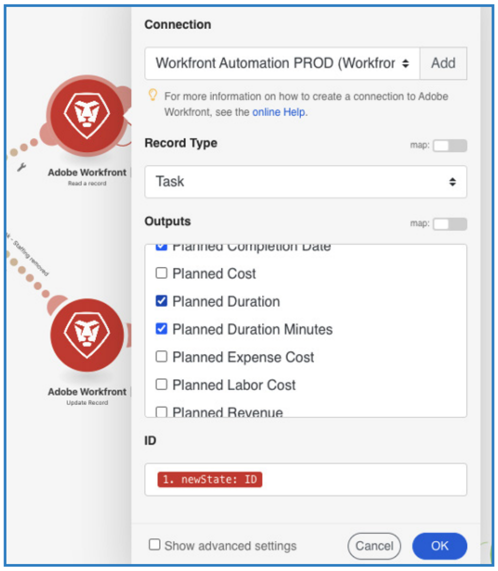


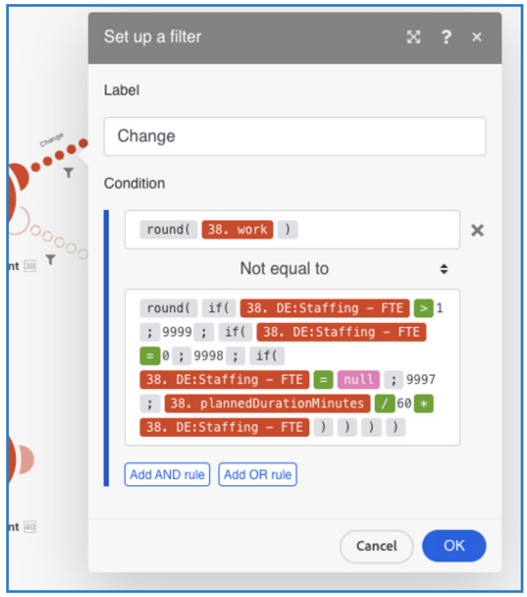



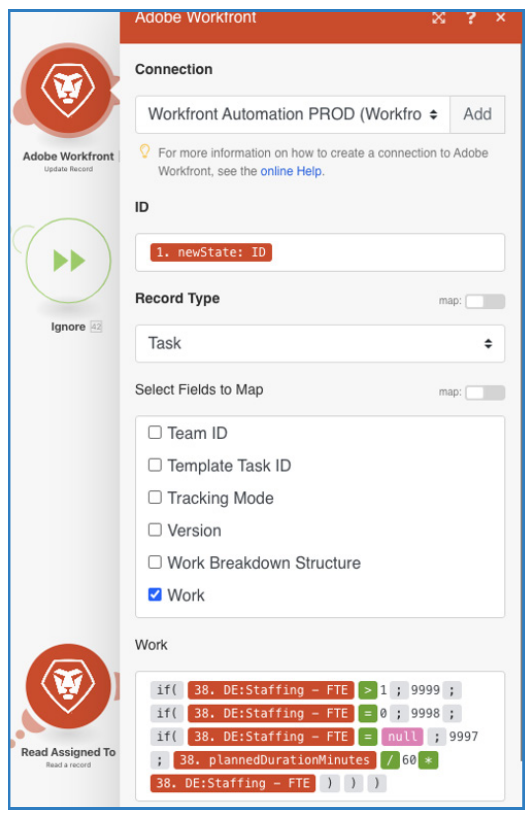
# Automating Data Attributes - Calculate Planned Hours

## Detailed instructions

**Step 3:** Read the task. Filter to verify that there is a change so that it doesn't update when there is no change.



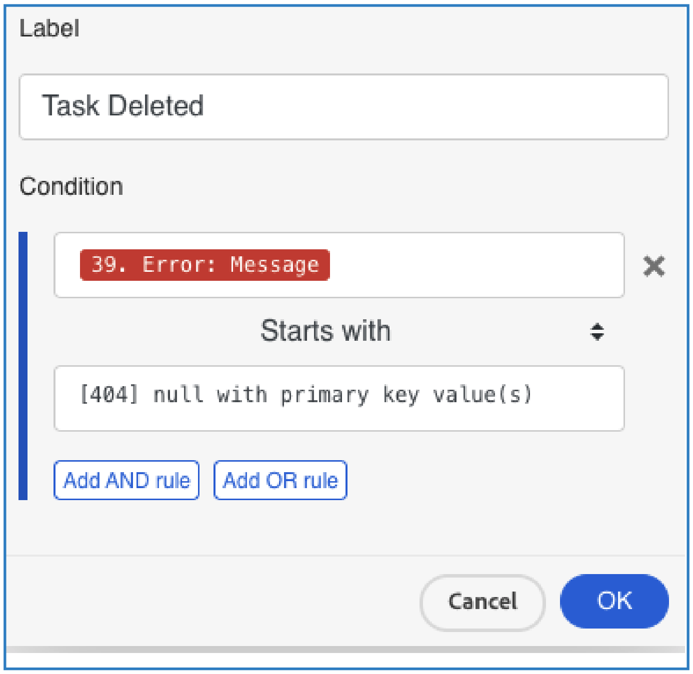

**Step 4:** Identify approach for flagging erroneous data. In this example, larger number outputs are used.

- **9999:** more than 1 FTE planned for a single person
- **9998:** FTE planned = 0
- **9997:** FTE planned is blank (most common user entry error)

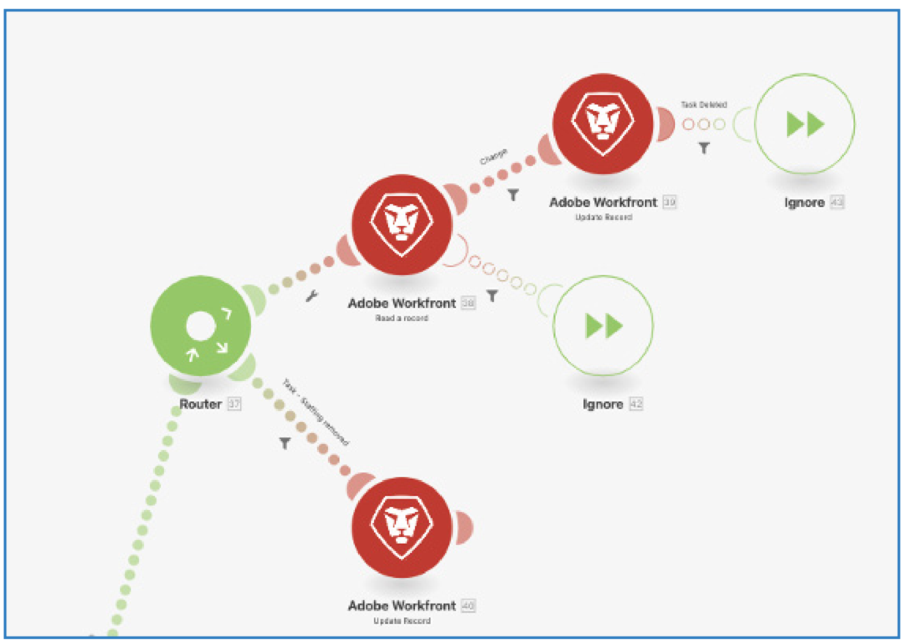
# Automating Data Attributes - Calculate Planned Hours

## Detailed instructions

**Step 5:** Add an error handler to ignore errors where the task is deleted.



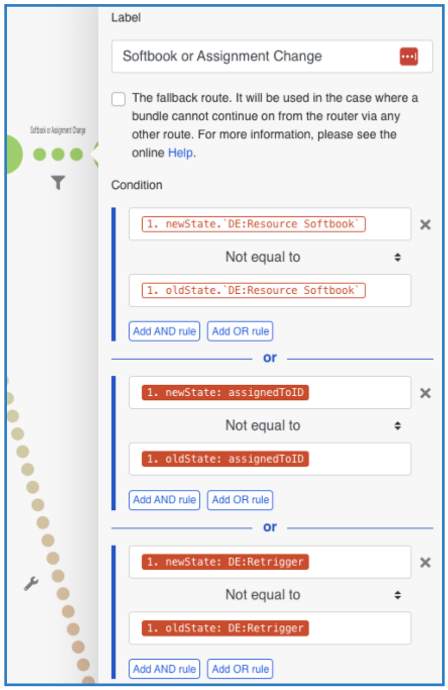

## Final product



# Automating Data Attributes - Manage Resource Calculated Reporting Fields

## Detailed instructions

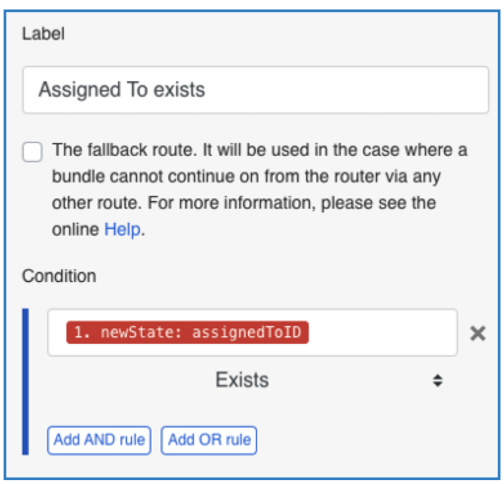
**Step 1:** Filter for relevant tasks - in this scenario, we are filtering for softbooks or assignment changes.

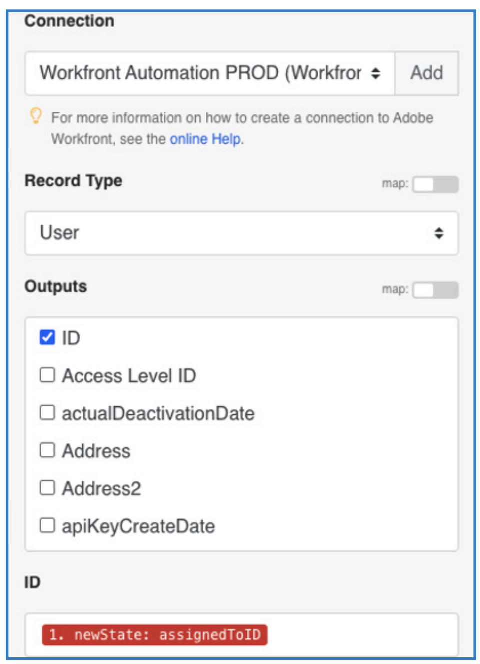



**Step 2:** Filters for Assigned To exist

**Step 3:** Read Assigned To record

- ID
- Name



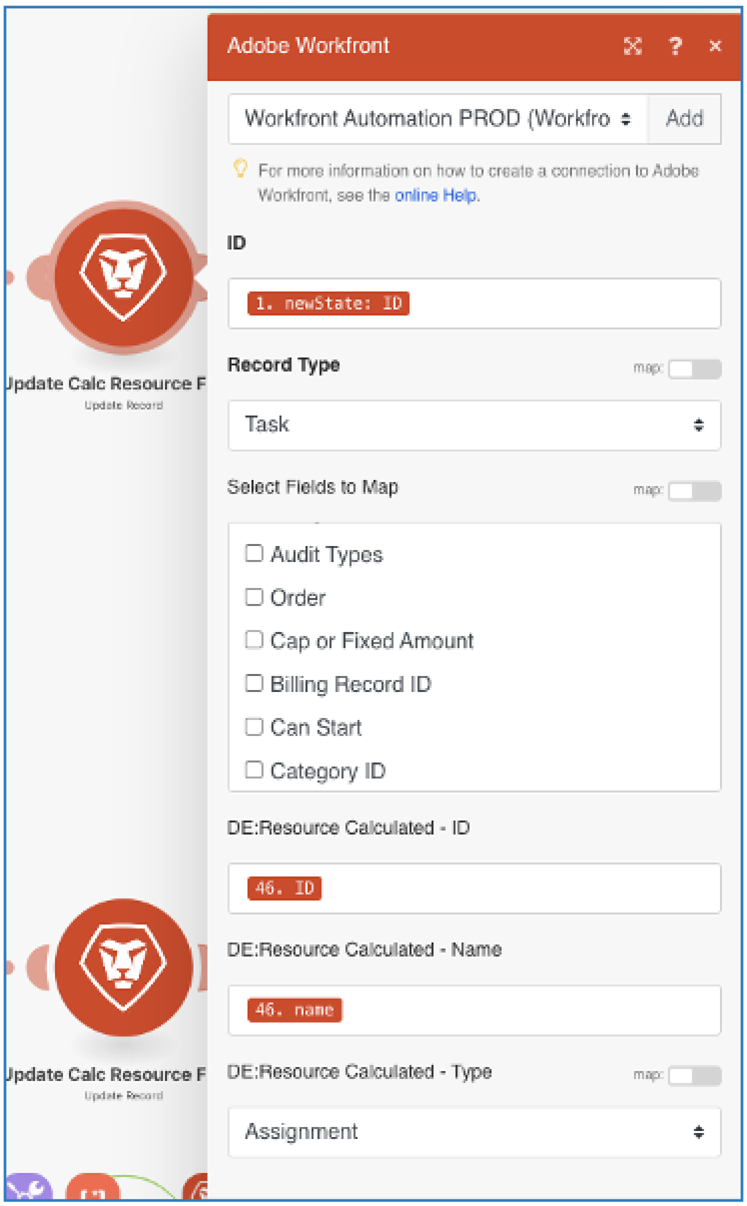



# Automating Data Attributes - Manage Resource Calculated Reporting Fields

## Detailed instructions

**Step 4:** Update project fields for Resource Calculated:

- Fields are used to support reporting and queries
- Provides a single field to pull from regardless if the user is assigned or softbooked
- Used in Capacity Charts



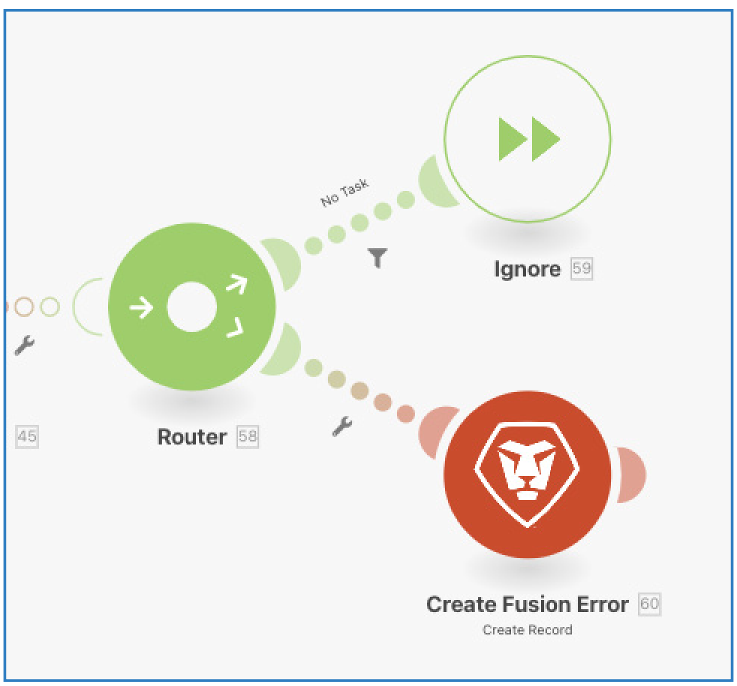
# Automating Data Attributes - Manage Resource Calculated Reporting Fields

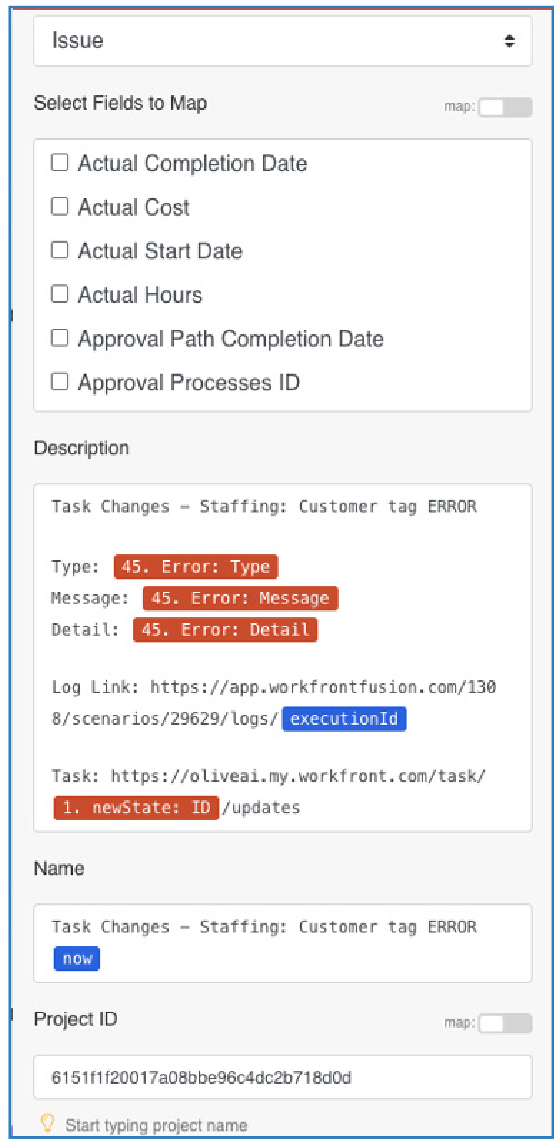
## Detailed instructions

**Step 5:** Set up an Error Handler with the following conditions:

- If the task is deleted: ignore
- If the task remains - create an issue in a Workfront project for tracking Fusion errors.

*Author's Note: We use this Fusion Errors project like a helpdesk queue for non-urgent errors that need to be reviewed but should not stop a scenario.*



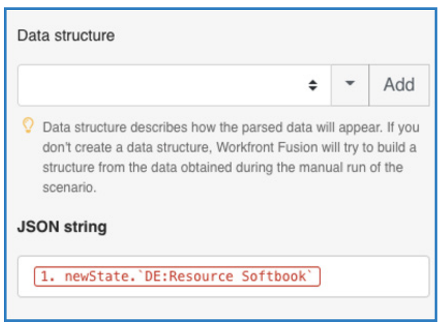
# Automating Data Attributes - Manage Resource Calculated Reporting Fields

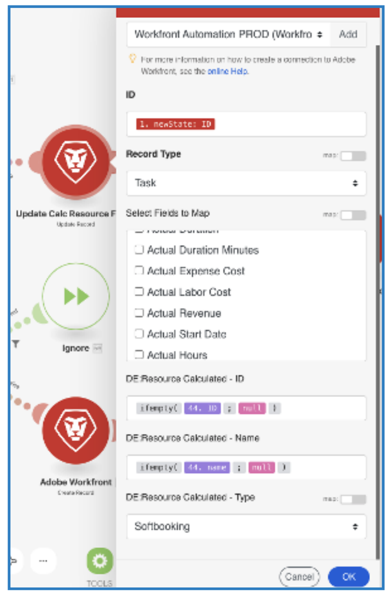
## Detailed instructions

**Step 6:** Set up a Softbook resource by:

- Parsing the field using JSON Parse
- Updating the Calculated Resource fields
- Setting up an Error handler

**Note:** Error Handler setup follows the same procedure as outlined in Step 5.



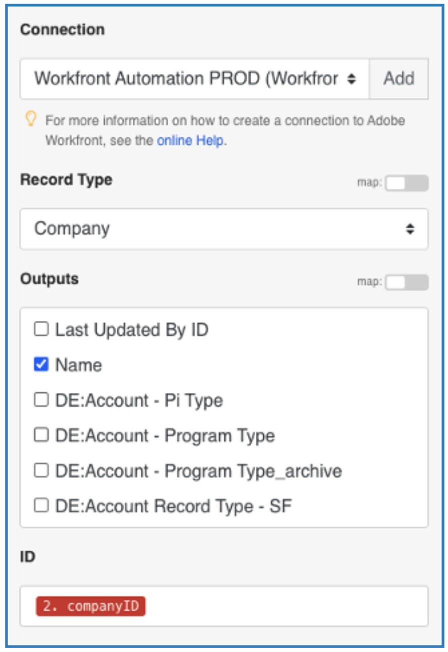




## Final product

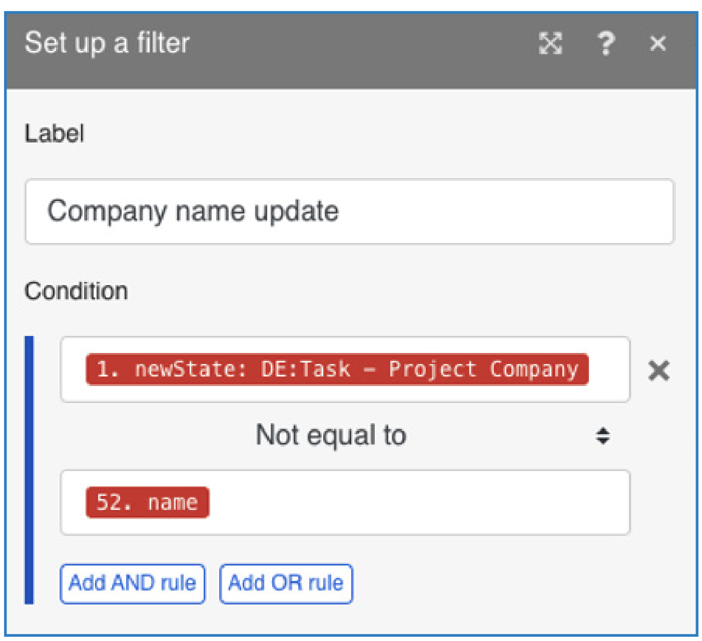
# Automating Data Attributes - Company Name on Staffing Task

## Detailed instructions

**Step 1:** Read Company Name or companyID as the output. If error on Company - create an Error ticket.



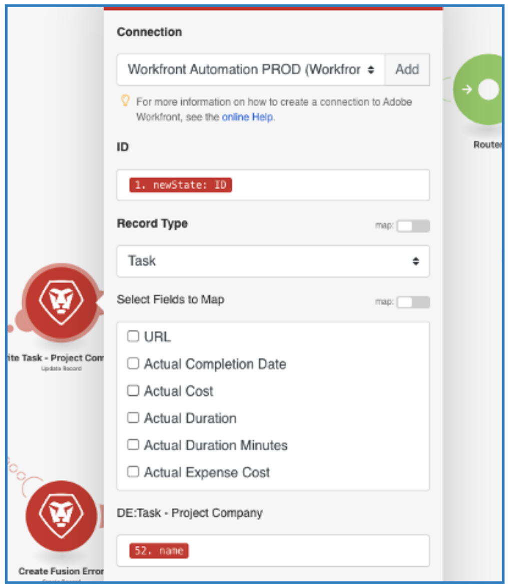

**Step 2:** Filter for company name change



# Automating Data Attributes - Company Name on Staffing Task

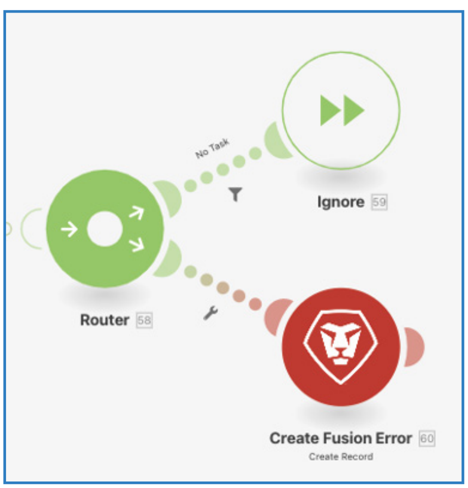
## Detailed instructions

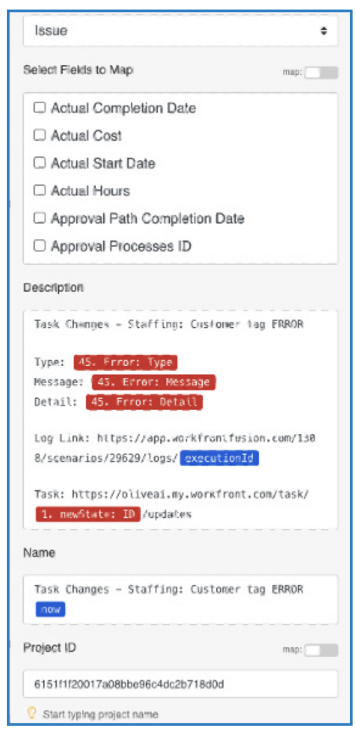
**Step 3:** Write company name in the custom field (used for reporting on task reports since it's too many jumps for WF reports).




**Step 4:** Set up an Error Handler.

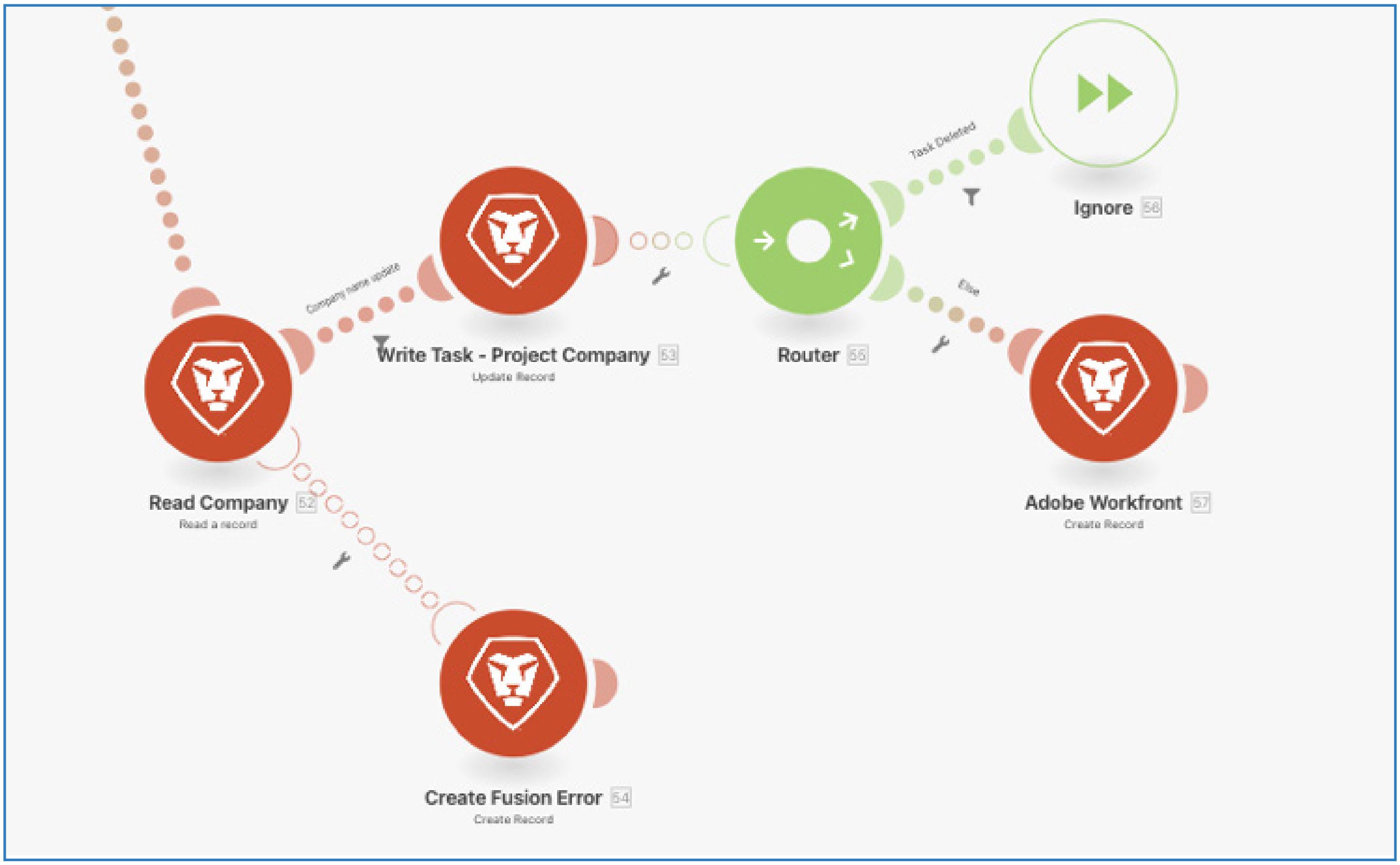
- If the task is deleted - ignore
- If the task remains - create a Fusion Error Record



# Automating Data Attributes - Company Name on Staffing Task

Final product



Adobe Workfront

FUSION REPORT RECIPES

# Automating Workfront Request Approval Workflows

**Prep:** 1-3 hours

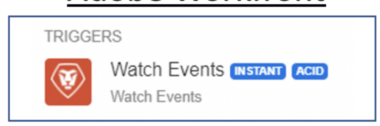
**Benefit:** Dynamically build complex request approval workflows based on custom field inputs

**Serves:** Traffic Managers, Business Requesters

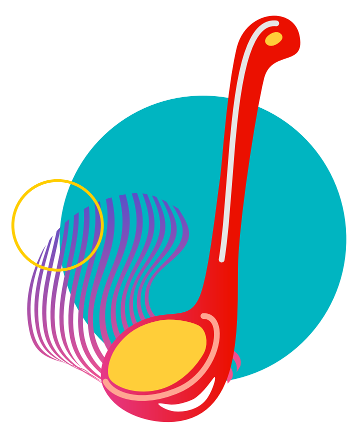
**Modules:** 1 Request Custom Form with desired custom fields and calculated fields

**By:** Veronica Bright

**API Modules:**
Adobe Workfront







# Automating Workfront Request Approval Workflows

## Detailed instructions

**Prep Work:**

1. **Map the Approval Process(es)**: Map out your business rules for the approval processes. Consider the rules as "IF / THEN" statements. For example, if a request is submitted by the "Retail" Line of Business and will cost more than $10,000, the Marketing Director and Marketing VP for the group need to approve it.

2. **Building the Calculated Fields**: Create a custom calculated field for each possible approval stage

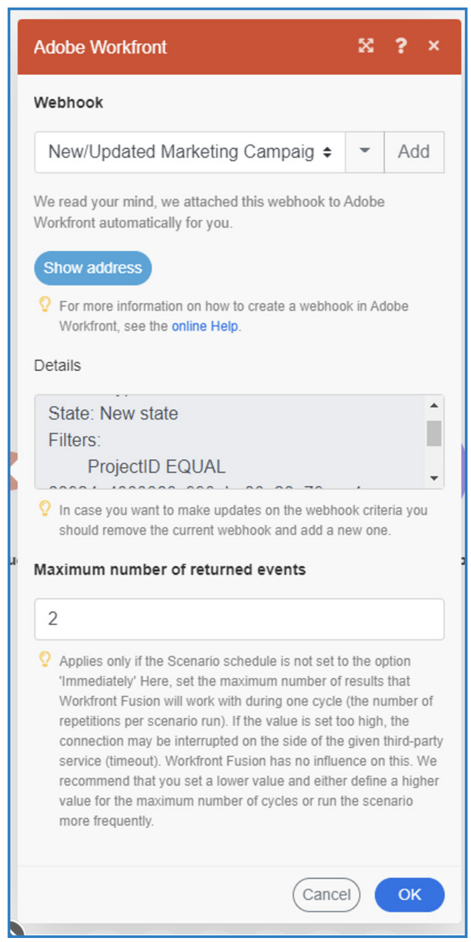
Use IF() statements in the field calculation to return a TRUE or FALSE value indicating if the request meets criteria to qualify for each approval stage.

3. **Setup User Profiles:** Dynamically find the right user for each stage by assigning the user a job role and group (EX: Role = Marketing Director, Group = Retail Line of Business). When a request requires Retail Marketing Director approval, this user will be assigned the approval step. User profiles must have the correct job role and home group so they can be "Found" by the automation.

**Step 1:** Use the Watch Events Module to listen for new issues added to Workfront. If the requests will only come in from a specific request queue, include the queue project ID in the filter to limit the number of events the webhook needs to process.

- Record Type: Issue
- State: New State
- Record Origin: New Records

Exclude Updates made by Connection User.

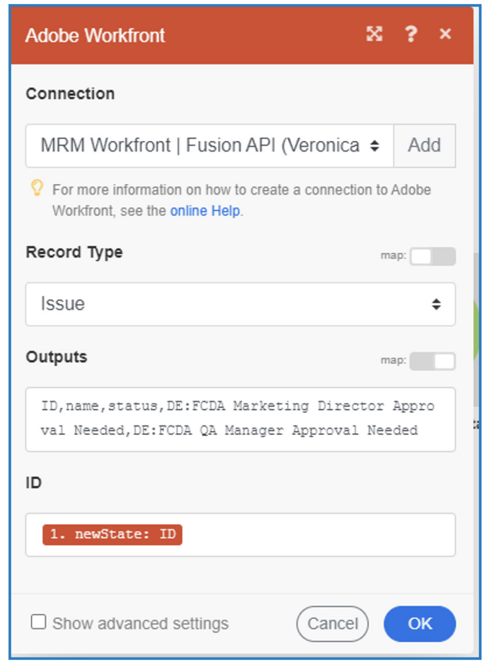



**Pro Tip:** Use a 'read record' module to extract any related fields instead of mapping them from the trigger module. If you need to change the trigger module in the future, you won't have to remap all the output fields.

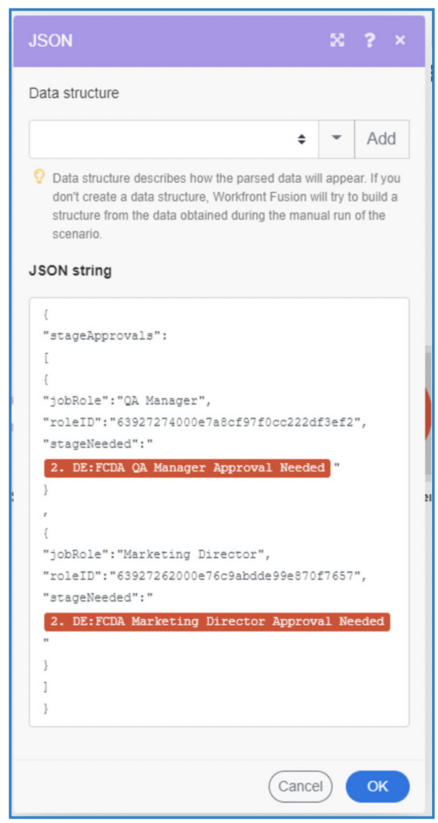
# Automating Workfront Request Approval Workflows

## Detailed instructions

**Step 2:** Add a **Read a Record** Module to read the Issue data, and extract the calculated fields indicating which stages are needed in the approval process from the custom form.



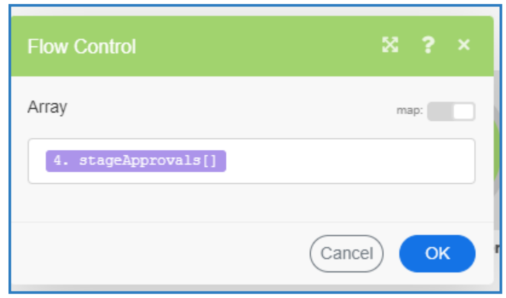

**Step 3:** Use a "Parse JSON" module to set the framework for the approval process(es).Here is where you will set the order of the approval stages (e.g. Marketing Manager reviews before Marketing VP) and use the calculated fields for each stage to indicate if the stage is required or not.



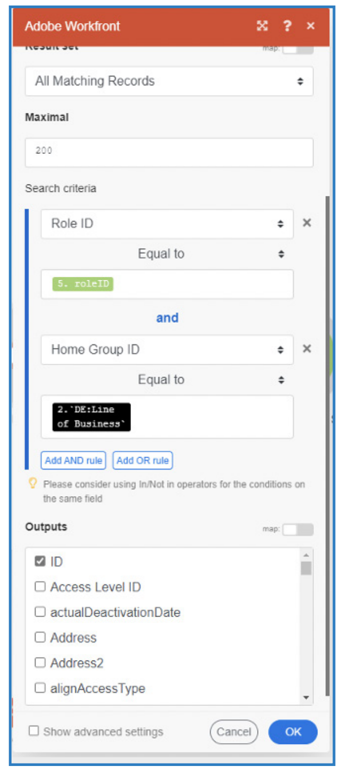
# Automating Workfront Request Approval Workflows

## Detailed instructions

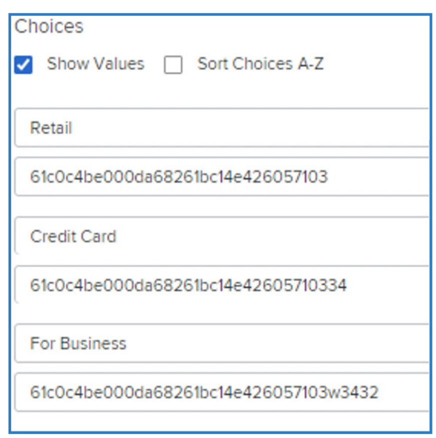
**Step 4:** Add an iterator module to iterate through the array of approval stages you identified from **Step 3**.




**Step 5:** Use a Workfront search module to search for the correct user to assign to the approval stage. The parameters will be based on your business process, but using the stage "role" and Line of Business "group" IDs is a great way to make sure the right person is assigned.



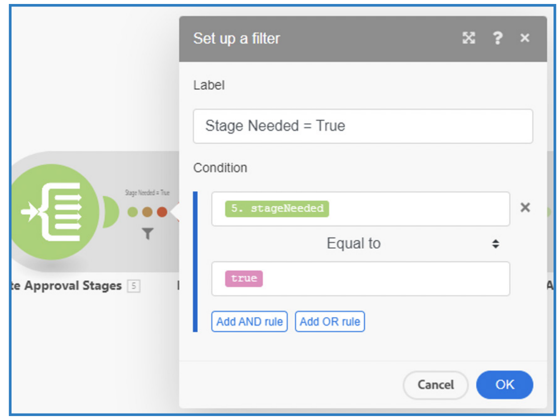

**Pro Tip:** In the custom field configuration within Workfront's UI, you can set the "value" of a drop-down field list values to a different back-end value that is passed into Fusion. For example, in our Line of Business custom field, we set the back-end value for the LOB list to match Group IDs that represent each Line of Business.



# Automating Workfront Request Approval Workflows

## Detailed instructions

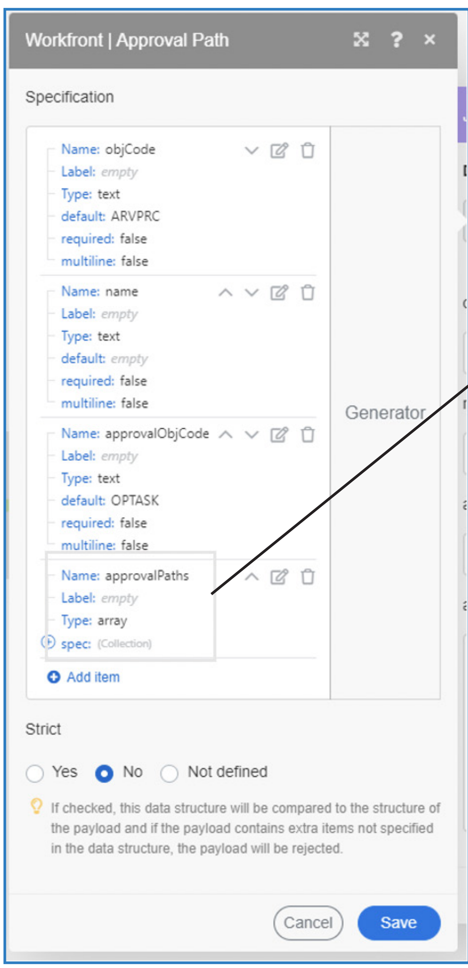
**Step 6:** Add a filter between the iterator module and the Search module in Step 5 to only filter stages where the parameter **"stage required" = TRUE**.



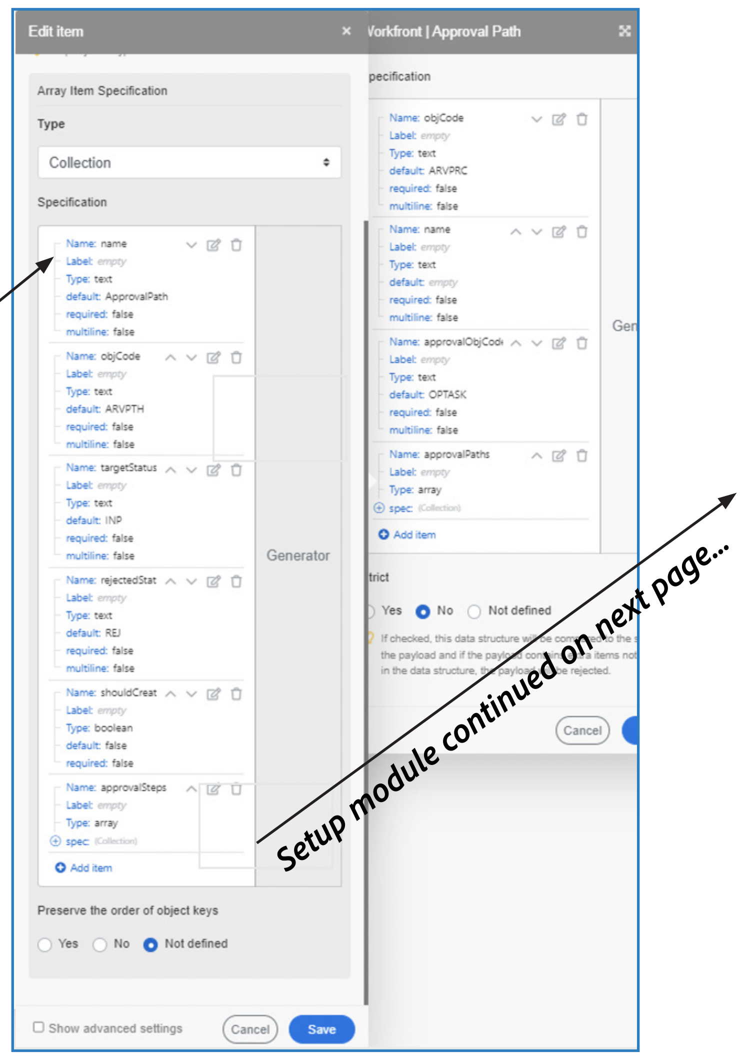

**Step 7:** Add a "Create JSON" Module with the Approval Process JSON data structure. This is a complex JSON structure with multiple complex arrays and sub-arrays.

The structure should look like this:

1. Top-Level Approval Inputs




approvalPaths Array: Collection Inputs

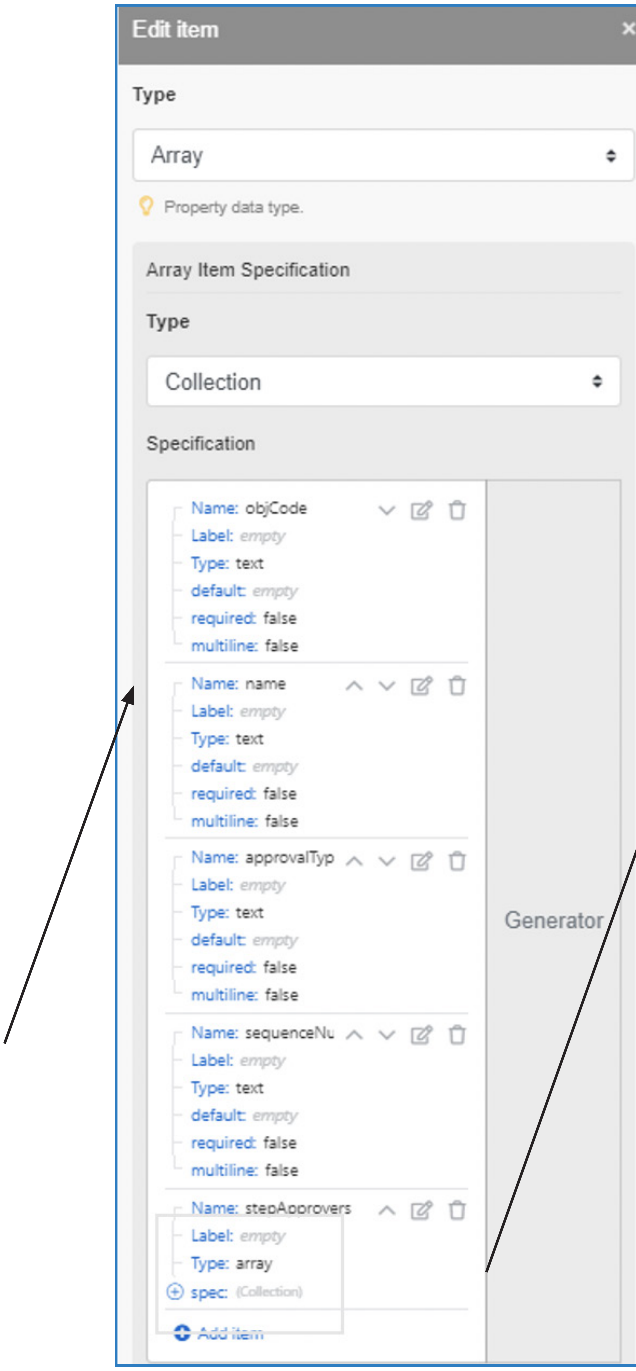


# Automating Workfront Request Approval Workflows

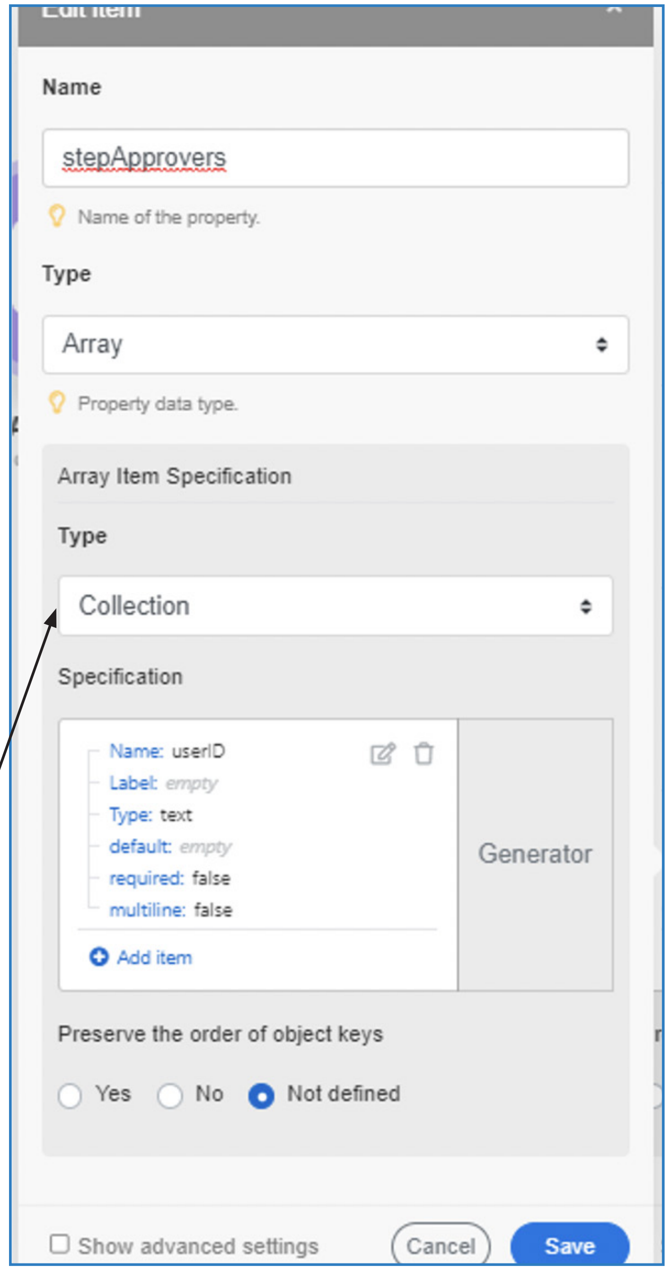
## Detailed instructions

**Step 7 (continued):** The structure should look like this:

**approvalSteps Array:** Collection Inputs



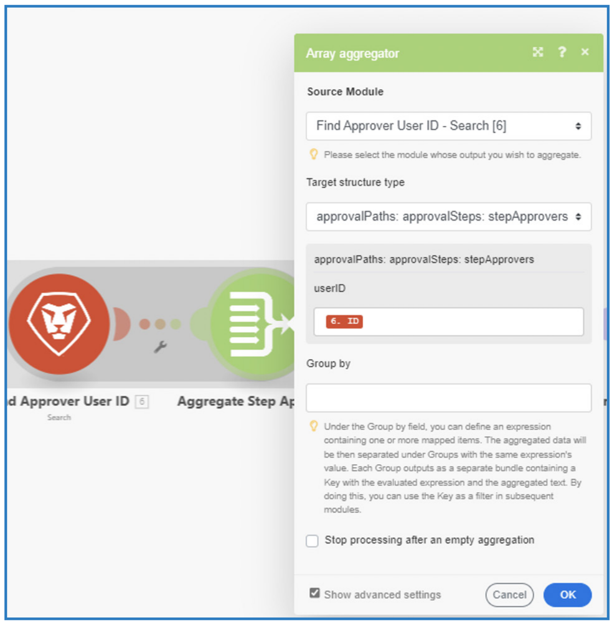

**stepApprovers Array:** Collection Inputs



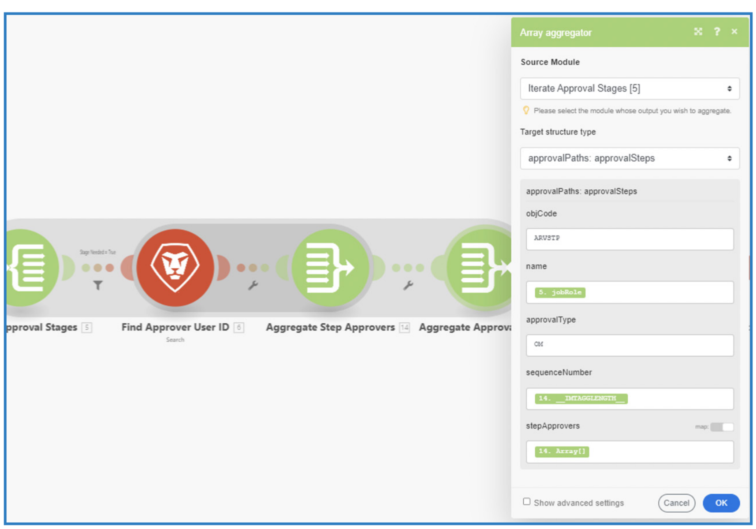
# Automating Workfront Request Approval Workflows

## Detailed instructions

**Step 8:** Add an Aggregate to Array Module BEFORE the "Create JSON" Module that you built in the previous step. Set the target structure type to approvalPaths:approvalSteps:stepApprovers from the 'Create JSON" Module. Map the user ID from the "Search" Module into the userID field. Set the Aggregator 'Source Module' to the "Search" Workfront Module.



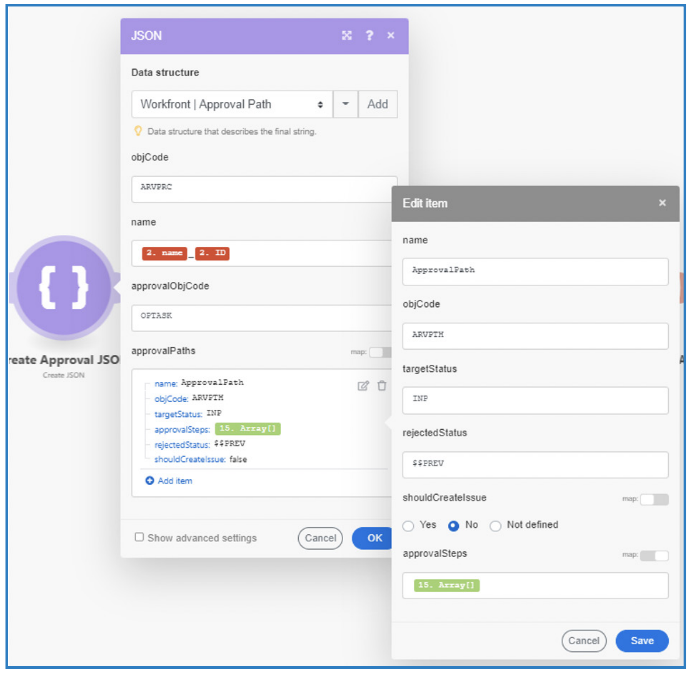

**Step 9:** Add another Array Aggregator Module BEFORE the "Create JSON" Module and AFTER the "Array Aggregator" from Step 9. Set the target structure type to approvalPaths:approvalSteps from the "Create JSON" Module. Map the output array from the stepApprover Array Aggregator into the stepApprovers field. Set the Aggregator 'Source Module' to the "Iterator" module.



# Automating Workfront Request Approval Workflows

## Detailed instructions

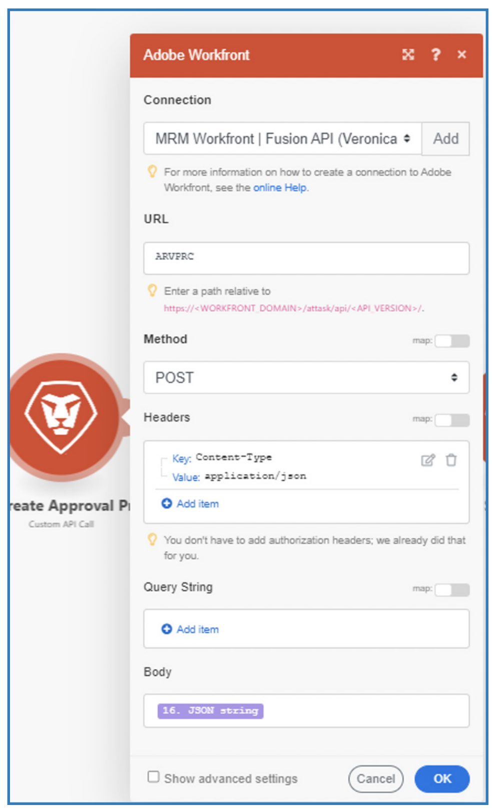
**Step 10:** Update the inputs to the “Create JSON” module that you built in the previous step with the following:




**Pro Tip:** Each approval will need a unique name, so use the “name” and “ID” from the request to set a unique name for this approval process.

Use the “map” toggle in the JSON inputs to map an array from a previous module into the array parameter.

**Step 11:** Use a Workfront “Custom API” Module to POST the ‘Create Approval JSON’ output to Workfront and create the new approval process.

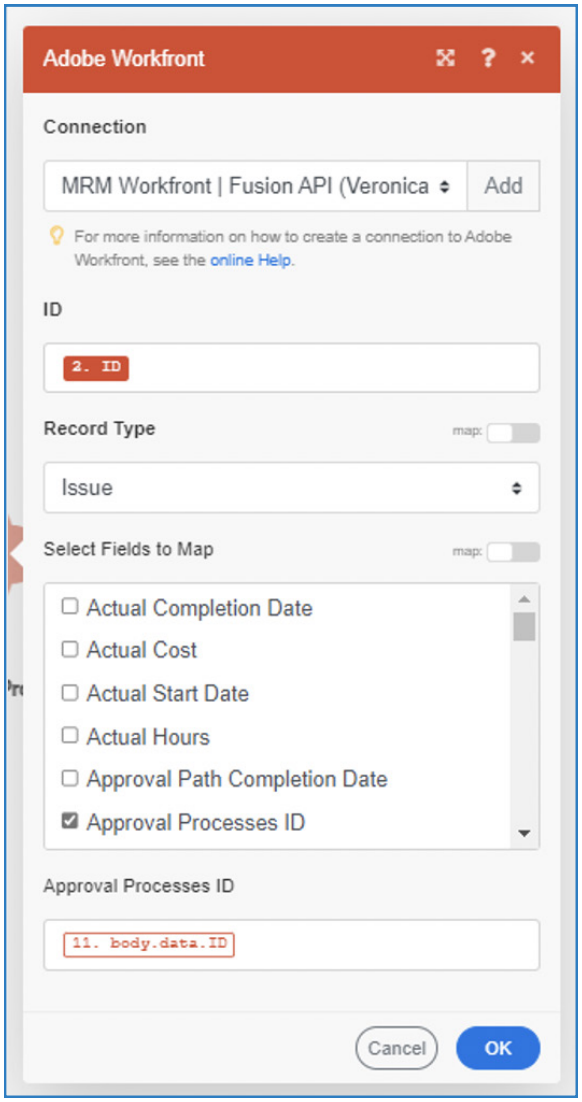


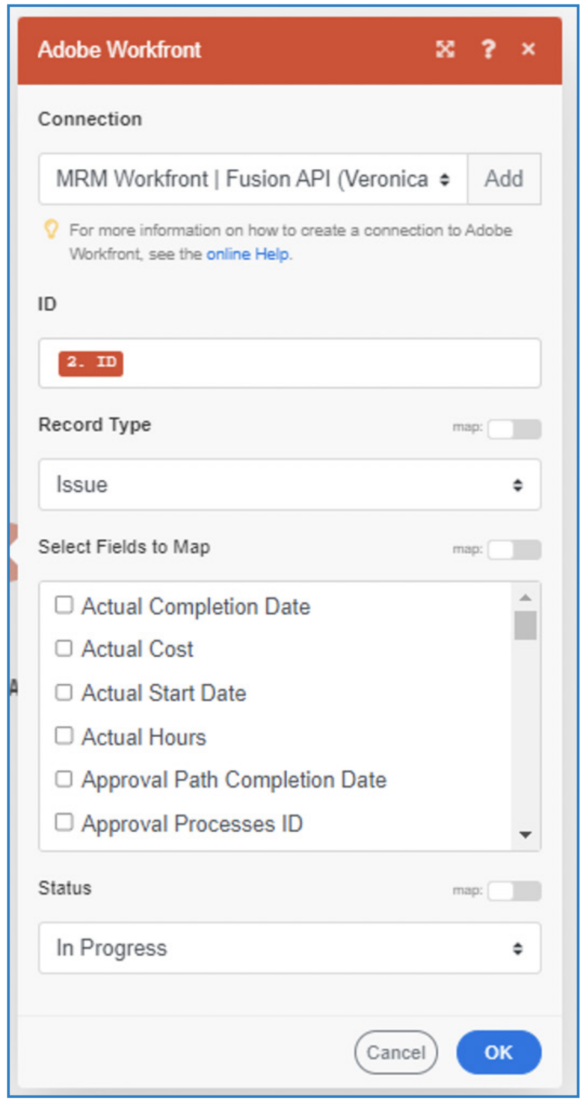

**Pro Tip:** In order to map outputs from one module to another, you may need to “Run” the output module first before the output fields show up in the mapping panel. This can be tricky to get sample data to “Run” a module, so you can use the following format to map the ID from the custom API call into the Update Record modules. Replace the ’11’ with the number of your API Call module: {{11.body.data.ID}}

# Automating Workfront Request Approval Workflows

## Detailed instructions

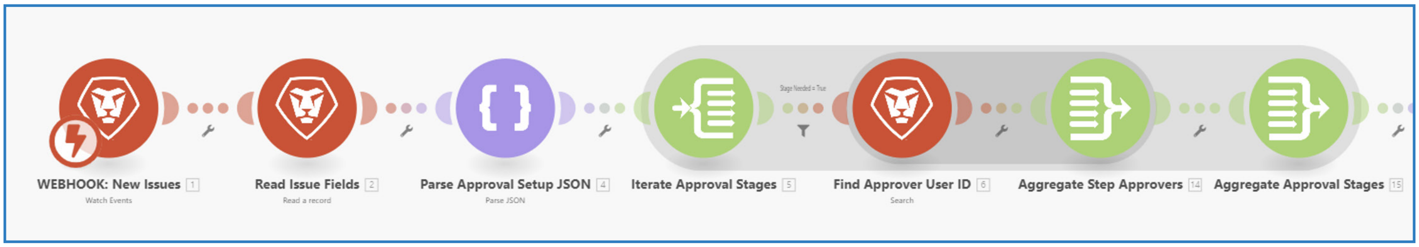
**Step 12:** Use an **"Update Record" Workfront Module** to Update the Request with the new Approval Process ID. Then use another "Update Record" Workfront Module to update the Request status to INP to trigger the approval process.

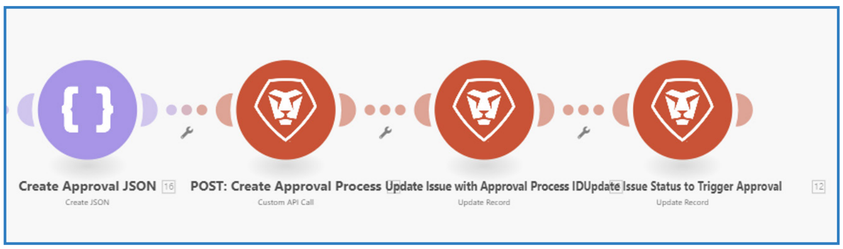


# Automating Workfront Request Approval Workflows

Final product



Adobe Workfront

Adobe

FUSION REPORT RECIPES

# Integrate Workfront to JIRA

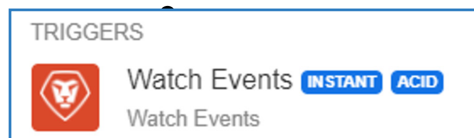
**Prep:** 15 minutes

**Yield:** Workfront projects represent initiatives that may need support from development teams who manage their work in Jira. The Workfront project needs to be linked to multiple Jira tickets on different backlogs, and the Jira status sync'd back to Workfront for tracking. Users can make multiple selections at the project level that automatically create the various Jira tickets needed and sync the Jira status and comments back to Workfront.

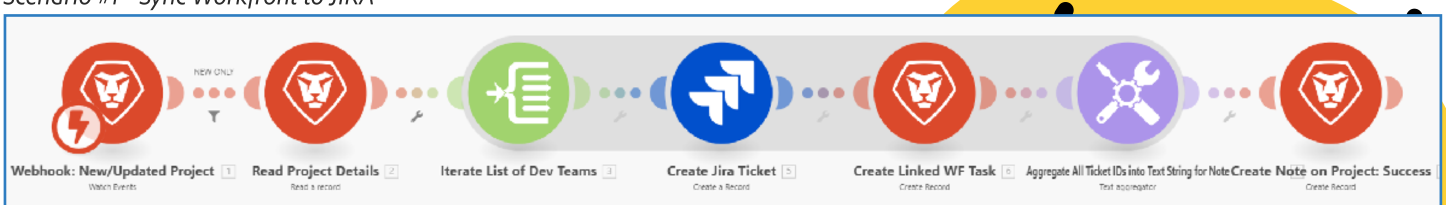
**Benefit:** Save time creating Jira tickets for dev teams to support projects in Workfront; Sync Jira data into Workfront for central tracking

**Serves:** Project Managers, Business Stakeholders, Execution/ Development Teams

**Modules:** Adobe Workfront, JIRA Cloud




*Scenario #1 - Sync Workfront to JIRA*




*Scenario #2 - Sync JIRA to Workfront*

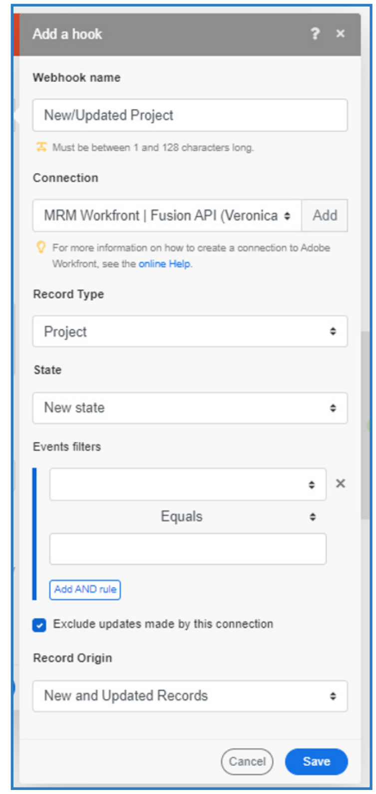
# Integrate Workfront to JIRA

*Scenario #1 - Sync Workfront to JIRA*

## Detailed instructions

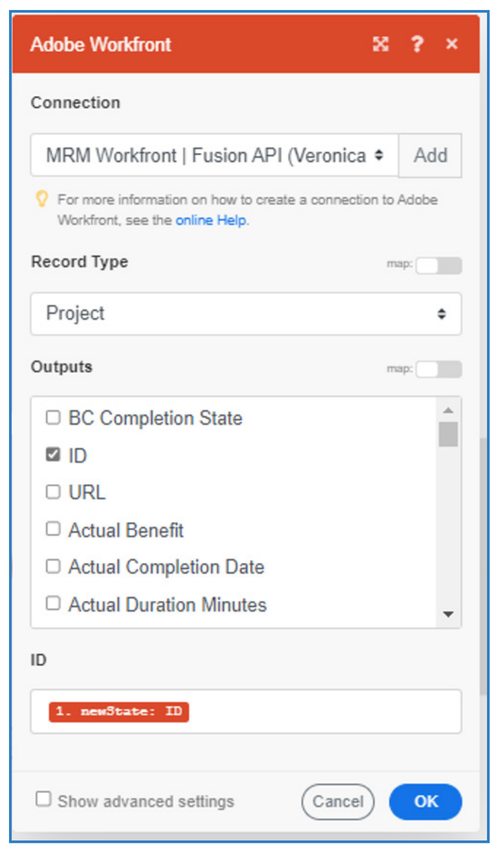
**Step 1:** Use the Watch Events Module to listen for new/updated Projects in Workfront

- Record Type: Project
- State: New State
- Record Origin: New and Updated Records




**Pro Tip:** To create multiple routes that ONLY proceed when a specific field has been updated, compare the newState value to the oldState value from the trigger module in a filter and only proceed if the values do not match.

**Step 2:** Add a Read a Record Module to read the project data, and extract the necessary fields from the Project custom form.

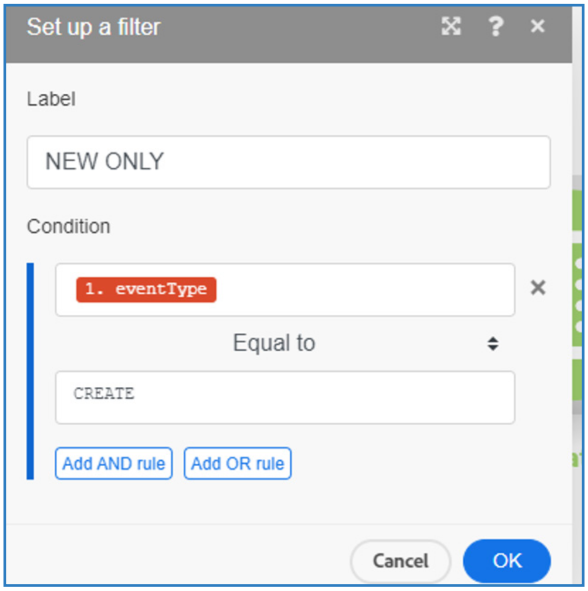


# Integrate Workfront to JIRA

*Scenario #1 - Sync Workfront to JIRA*

## Detailed instructions

**Step 3:** Add a filter between the trigger module and the Read Record module so only NEW projects will continue through the scenario.




**Pro Tip:** Jira Cloud requires a user's "account ID" as a required value in the 'Reporter' field when creating an issue.

You could set a default reporter by entering the ID of the service user, or create a mapping table using a data store.

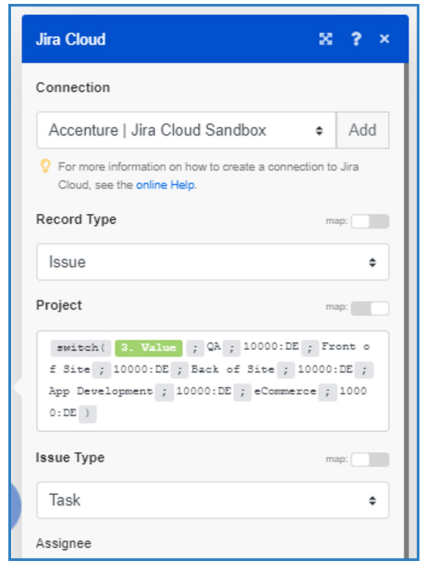
**Step 4:** Use an iterator on the multi-select custom field(s) to process the list of tickets that are needed. In this case, only the 'Development Teams' custom field will drive the tickets that are created.

# Integrate Workfront to JIRA

*Scenario #1 - Sync Workfront to JIRA*

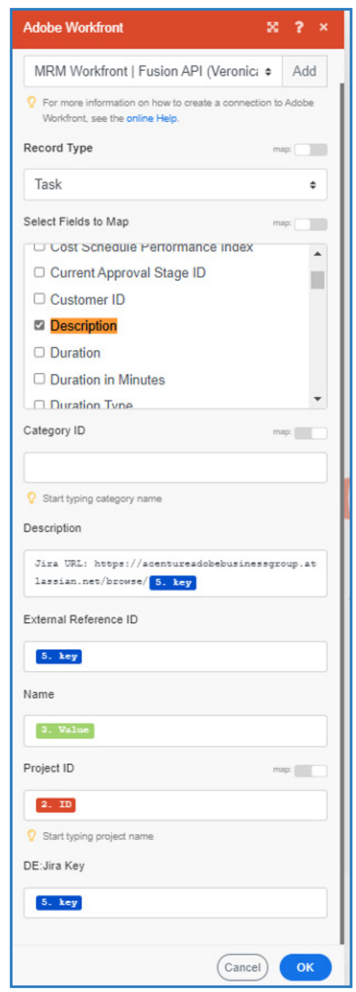
## Detailed instructions

**Step 5:** Use the “Create Record” Jira module to create the Jira issue. To create a ticket on each team’s Jira project, use a switch() function to switch the project ID based on the development team that will work on the ticket.




**Pro Tip:** Always include the custom form ID, AKA the “Category ID” when updating custom fields on Workfront objects

**Step 6:** Use a “Create Record” Module to create a linked task on the originating Workfront Project where the Jira status and comments can be tracked. The Jira Key can be captured in a custom field or in the Workfront system field ‘External Reference ID’

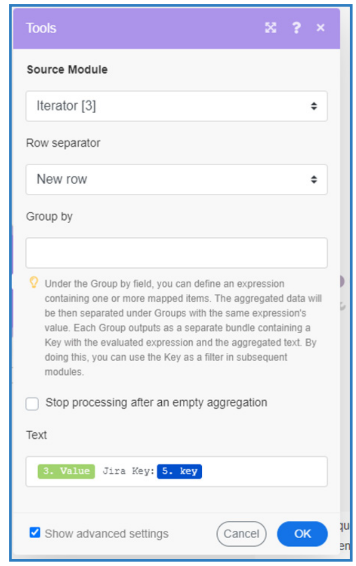


# Integrate Workfront to JIRA

*Scenario #1 - Sync Workfront to JIRA*

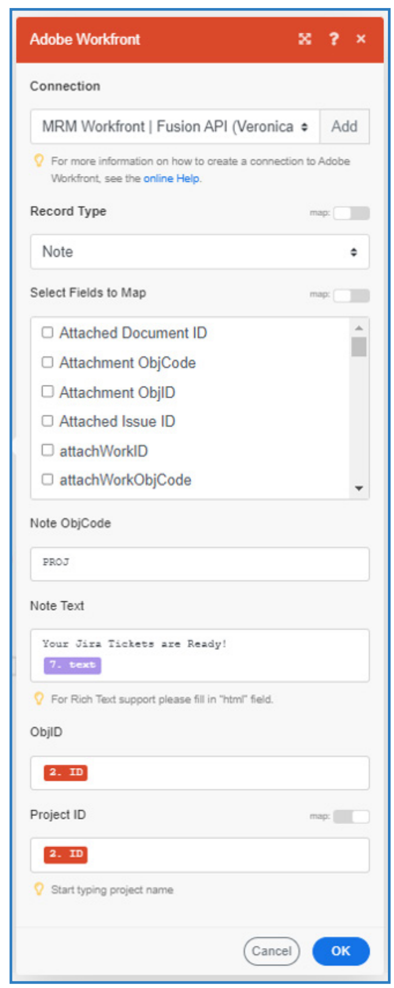
## Detailed instructions

**Step 7:** Use a text aggregator to aggregate all the Jira ticket keys into a text string, and aggregate the iterator bundles into a single bundle for the final step. Use a “New Row” as the Row Separator for Readability.




**Pro Tip:** To make this scenario more robust, add a router to the beginning of the scenario and have Fusion complete different steps when a NEW project is being “Created” or an EXISTING project is being “Updated”

**Step 8:** Use the “Create Record” Module to create a Note on the project informing the project owner that the Jira tickets are complete.

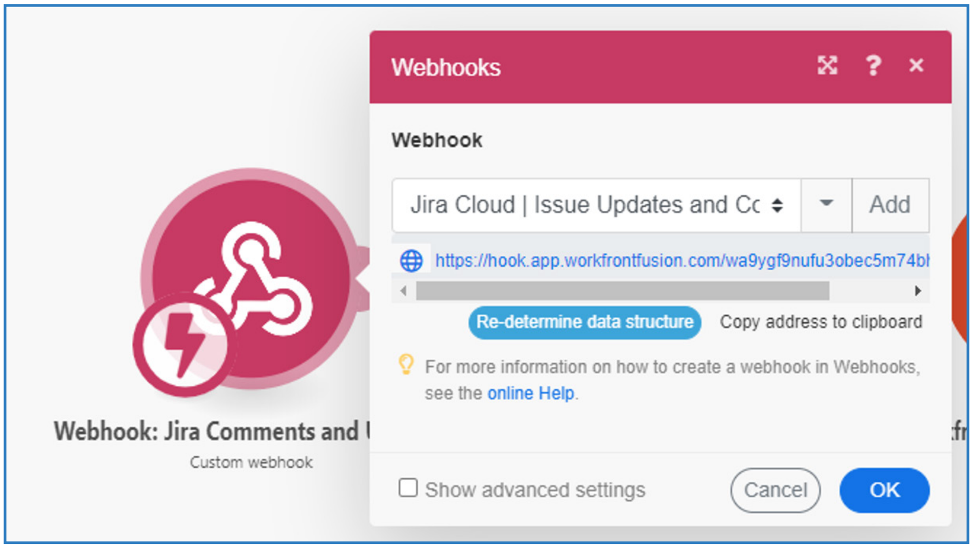


# Integrate Workfront to JIRA

*Scenario #2 - Sync JIRA to Workfront*

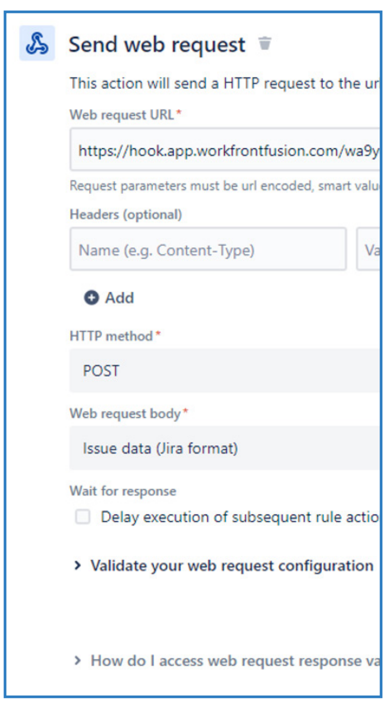
## Detailed instructions

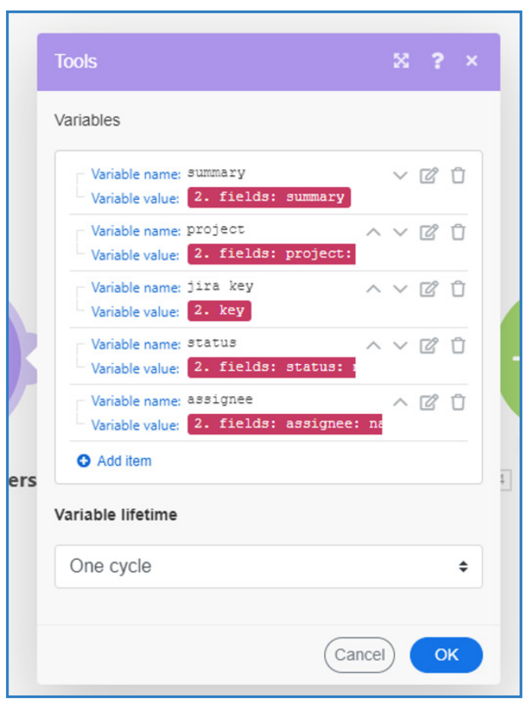
**Step 1:** Use a webhook module to capture the data feed from JIRA.




**Pro Tip:** You can use the same webhook URL to configure the Jira data feed from multiple Jira projects AND for multiple Jira trigger events (New Comments, Updated Issues, Assigned Issues, etc.) to reduce the number of redundant scenarios.

**Step 2:** Use a set variable module to set the JIRA Issue parameters from the Webhook. When configuring the automation settings in Jira, select “Issue Data” in the Web Request Body.



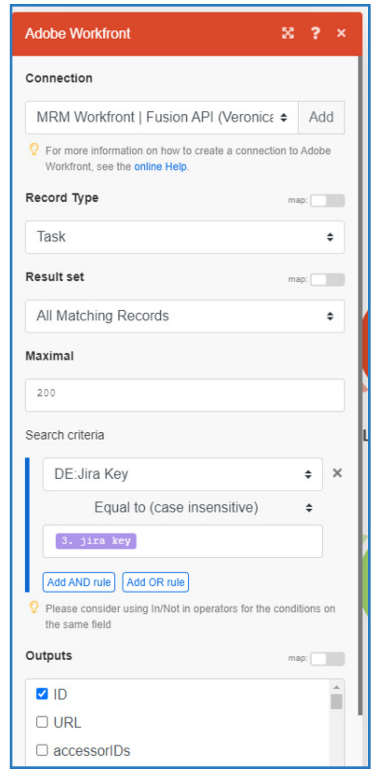



# Integrate Workfront to JIRA

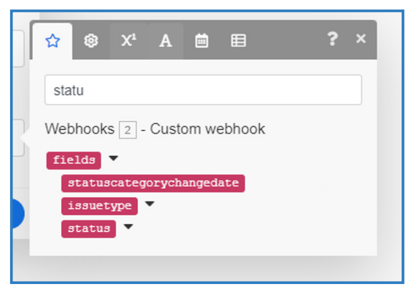
*Scenario #2 - Sync JIRA to Workfront*

## Detailed instructions

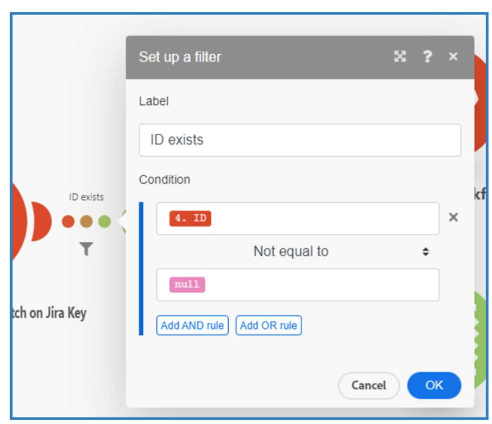
**Step 3:** Use a Workfront Search module to search for Workfront tasks where the DE:Jira Issue Key (or whichever field you set the Issue Key to populate from Scenario #1) = Issue Key from JIRA.




**Pro Tip:** Use the search bar in the mapping panel to quickly find and map fields from previous modules.




**Step 4:** Add a router, then before the router add a filter to only proceed if a matching task record is found from the Search module.



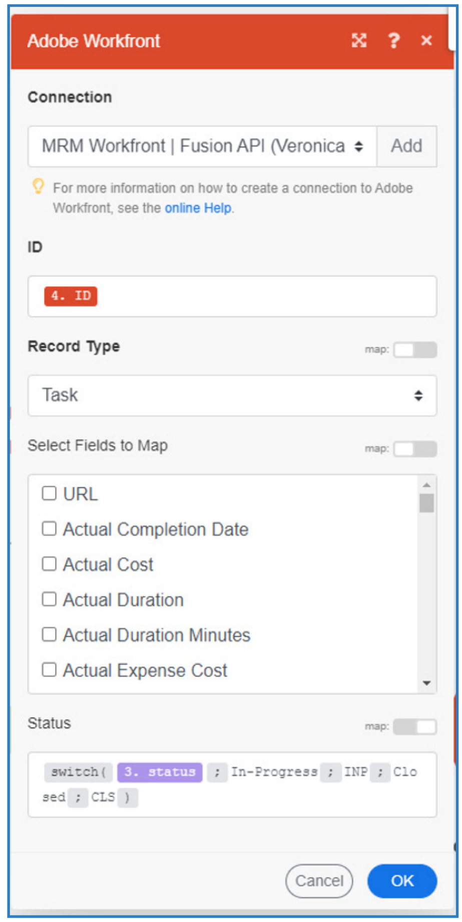
# Integrate Workfront to JIRA

*Scenario #2 - Sync JIRA to Workfront*

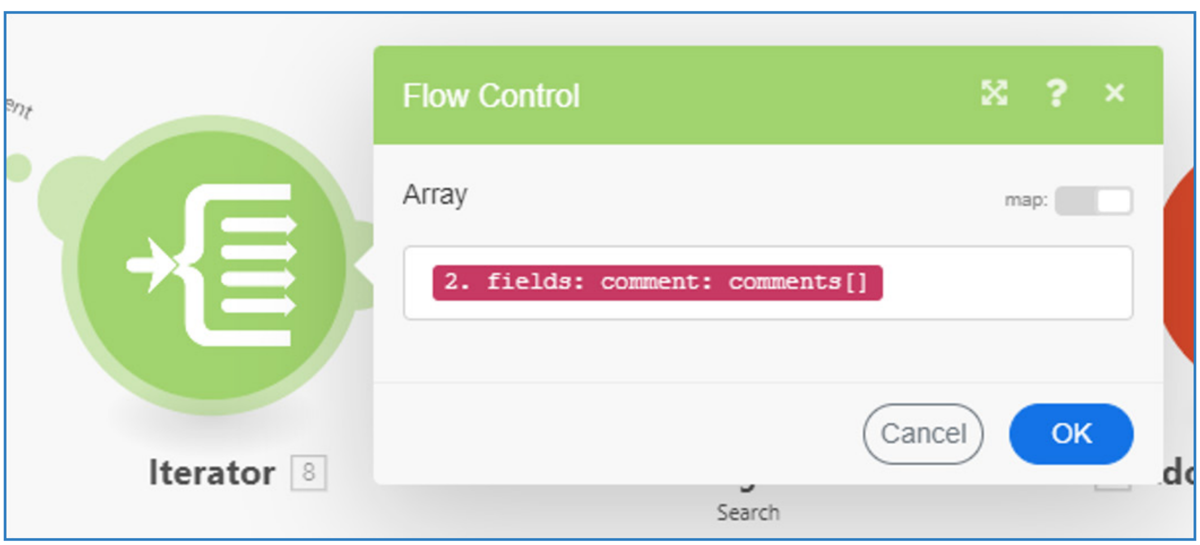
## Detailed instructions

**Step 5:** Add an "Update Record" Workfront module to the first route of the router. Map the fields from the Jira ticket that should be updated in Workfront.

**Pro Tip:** Use a "Switch" function to swap the Jira status for the Workfront task status code in the field mapping




**Step 6:** Add an Iterator Module to the second route of the router, and map the "Comments []" array from the webhook into the iterator.

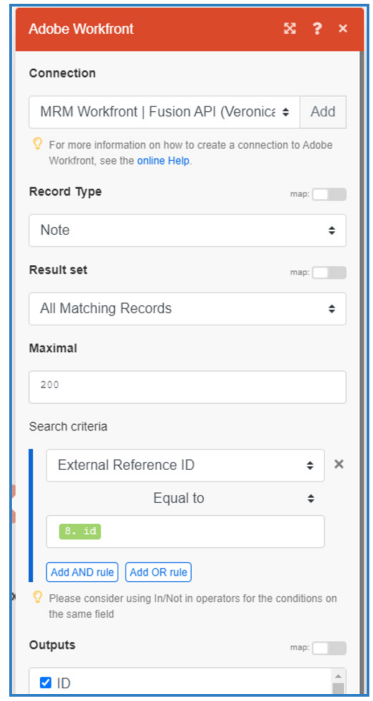


# Integrate Workfront to JIRA

*Scenario #2 - Sync JIRA to Workfront*

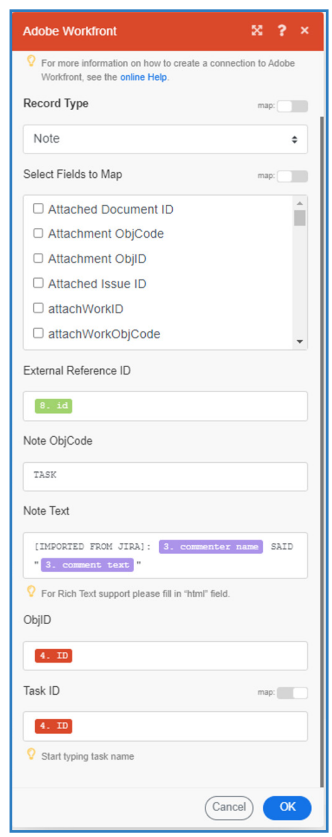
## Detailed instructions

**Step 7:** Add a “Search” Workfront module after the iterator to search for Notes where the ‘External Reference ID’ equals the Jira Comment ID.




**Pro Tip:** When using Workfront search modules (and most other application’s search modules), make sure to select at least one output field or you will receive an error

**Step 8:** Add a “Create Record” Module to create a new Note with the Jira Comment inputs. Make sure to set the ‘External Reference ID’ value on the Workfront Note with the Comment ID from Jira so it doesn’t get duplicated.

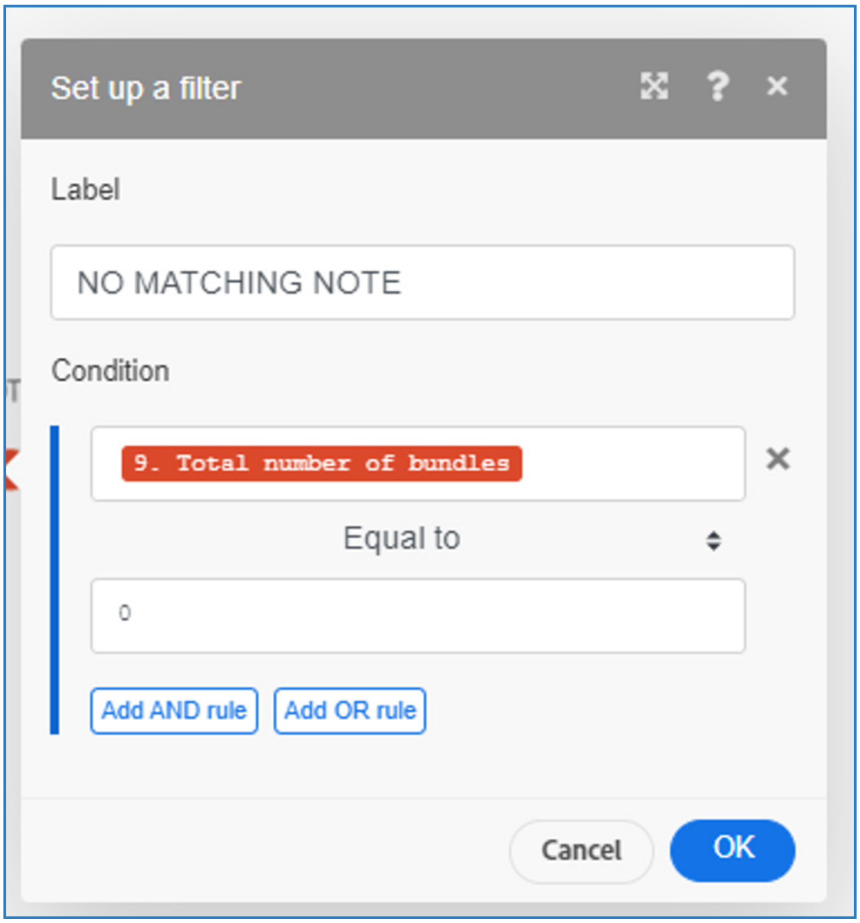


# Integrate Workfront to JIRA

*Scenario #2 - Sync JIRA to Workfront*

## Detailed instructions

**Step 9:** Add a filter between the Note "Search" and the Note "Create Record" module so a note is only created when there is not already an existing note.



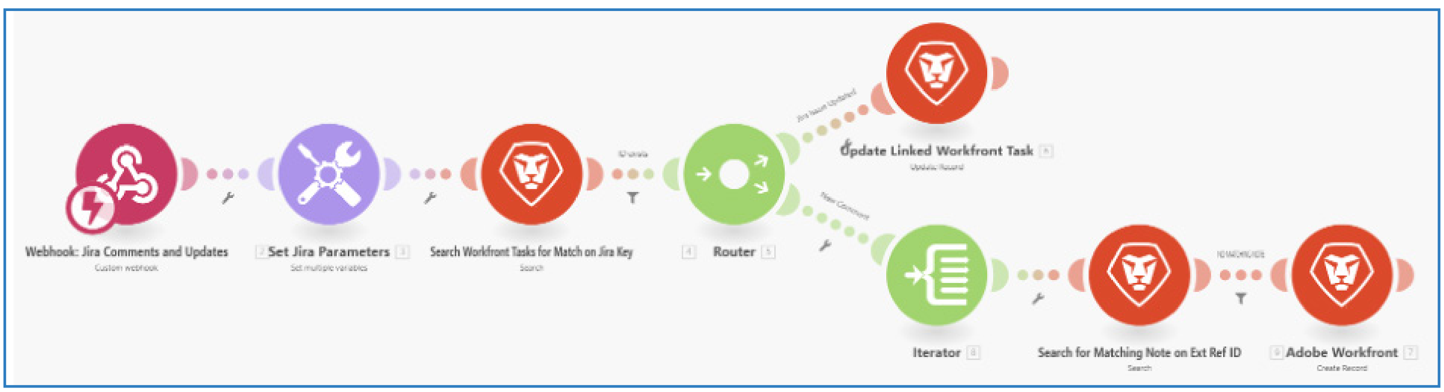

**Pro Tip:** When setting conditions in filters, make sure you are using the correct data type operator (array, text, date, etc.) to avoid the filter inadvertently blocking records from passing through

# Integrate Workfront to JIRA

## Final product

*Scenario #1 - Sync Workfront to JIRA*


*Scenario #2 - Sync JIRA to Workfront*



# That's a wrap for the Fusion Cookbook!

We hope you enjoyed these recipes and look forward to seeing you try them in your own kitchens.

**Questions or Feedback?**

*Let us know in the comment section below, and be sure to tag the recipe author or Community Manager!*