Unified Knowledge Documentation

How to Add GitHub Markdown as a Unified Knowledge Source

14 August 2024

This Document Is Currently A Beta Version. Please Report Any Issues Using The Feedback Button.

Contents

GitHub Markdown as a Unified Knowledge Source

Connecting a GitHub Repository to Unified Knowledge allows you to transform the markdown content in that repository to read-only knowledge article objects inside Salesforce. Any addition, modification, or deletion of content in the repository will be continuously and automatically reflected in Salesforce. Once in Salesforce, those knowledge articles can be natively viewed and searched, and grounded on by Einstein.

GitHub and Unified Knowledge High Level Overview

Workflow

1. Prepare Your Salesforce Environment for GitHub Markdown Content
2. Configure GitHub Markdown as a Unified Knowledge Source
3. Prepare Your GitHub Repository
4. Syncing and Monitoring GitHub Markdown Content Uploads

Requirements and Limitations

GitHub

- **File Type**: Markdown files (.md) and associated image files which are referenced from the markdown file are supported and will upload. All other files will not.

> **Note:** Content that cannot be uploaded is ignored.

Salesforce

- **Number of Articles**: Based on your subscription edition in Salesforce, there is a limitation to the number of articles your knowledge base can have. Learn more about knowledge base allocations in Salesforce Help, you can contact your Salesforce account executive to increase your allocation.
- **Knowledge Field Character and Size:** There is a limitation of **131,072 characters that a Salesforce text area (Rich) field can contain**. If you exceed this limitation, your content may upload without styling. The following character and size limitations are applied to the Salesforce knowledge field.

  Regular knowledge field: 131,072 characters, a size limit is not applied. If an article has inlined images (e.g. base 64 embedded into the HTML), these images are not considered as attachments, and count towards the character limitation.

  Large content storage field: 4MB, a character limit is not applied. If an article has inlined images (e.g. base 64 embedded into the HTML), these images are not considered as attachments, and count towards the 4MB limitation.

  Regular field or large content storage field with attachments: 20MB.

  The default setting is to have the regular knowledge field. If you want to implement the Large Content Storage Field, contact your Salesforce account executive.
- **Internal Linking**: Unified Knowledge doesn't currently support cross references between two KB Articles. If your content creates more than one KB Article, your cross references will most likely break.

Prepare Your Salesforce Environment for GitHub Markdown Content

This guide explains how to add a GitHub repository as a content source to Salesforce Unified Knowledge. Once you set up your repository as a Unified Knowledge source, any supported file located in the shared repository can be uploaded to Salesforce automatically.

Configuration Information for Unified Knowledge

The following settings should be configured by your organization's Salesforce Administrator. They will be used to fill in the required fields for setting up the connection with Unified Knowledge.

Knowledge Field

1. In Salesforce **Setup**, go to **Object Manager**, select the **Knowledge** object, and create a **Text Area (Rich)** field type of 131,072 characters. Your source content will be uploaded to this Salesforce field. Example: ExternalBodyContent__c The character limitation is applied after the content has been uploaded to Salesforce. If you've enabled the Large Content Storage for Unified Knowledge feature for your org, a new field named **ArticleContent** will be added to the Unified Knowledge Source Configuration **Knowledge Field** dropdown, as shown here.

2. In Salesforce, go to **Page Layouts**, and add the field you created to the relevant page layout. If you've enabled the **Large Content Storage for Unified Knowledge** feature for your org, this field will show.

FIELD LABEL	FIELD NAME	DATA TYPE
Article Content	ArticleContent	Rich Text Area(4194304)

Note: If you upload more than one content source to Unified Knowledge, you can set the knowledge field once and apply it to all of your sources. Alternatively, you can set it individually per source.

Record Type

Create record types indicating the different formats of your knowledge articles which will be uploaded to Unified Knowledge. Examples: Google Drive, Jira Cloud, Microsoft Word.

Data Category

If your organization already has data categories in Salesforce, you can use them to map content labels between your source and Salesforce. If you don't have data categories in Salesforce, and want to map your source content labels to Salesforce, work with your content manager to map your source content labelling to new Salesforce data categories in Salesforce. Alternatively, you can choose not to label your content. Learn more about data categories in the Salesforce help center.

Next Steps

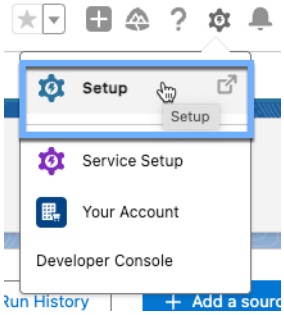
- Prepare Your GitHub Repository and Content
- Configure GitHub Markdown as a Unified Knowledge Source
- Syncing and Monitoring GitHub Markdown Content Uploads

Configure GitHub Markdown as a Unified Knowledge Source

In this document you can learn how to add a GitHub repository as a content source to Salesforce Unified Knowledge. Once you set up a GitHub repository as a Unified Knowledge source, any supported file located in the repository can be uploaded to Salesforce automatically or manually.

Configure Your GitHub Repository as a Unified Knowledge Source

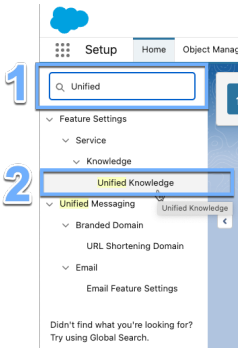
1. From the Salesforce main dashboard, click the cogwheel to open the **Settings** and select **Setup**.



2. In the left navigation panel:

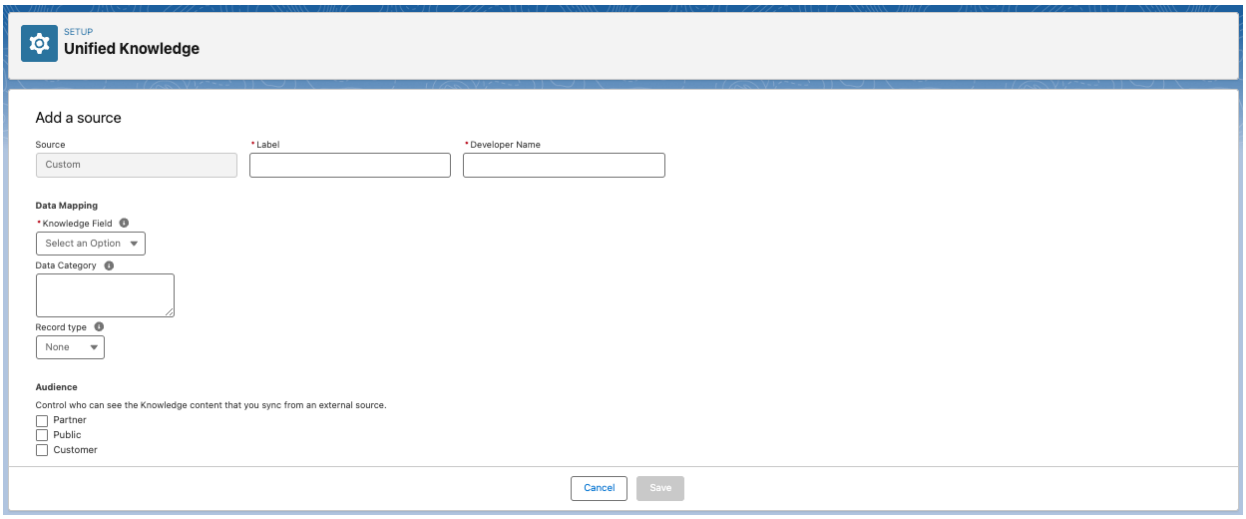
    1. In the **Quick Find** search, search for **Unified Knowledge**.

    2. Select **Unified Knowledge** from the search results.

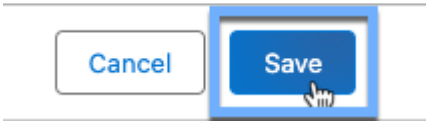


3. The **Unified Knowledge Dashboard** opens. Under **Unified Knowledge Sources**, click **+ Add Source**.

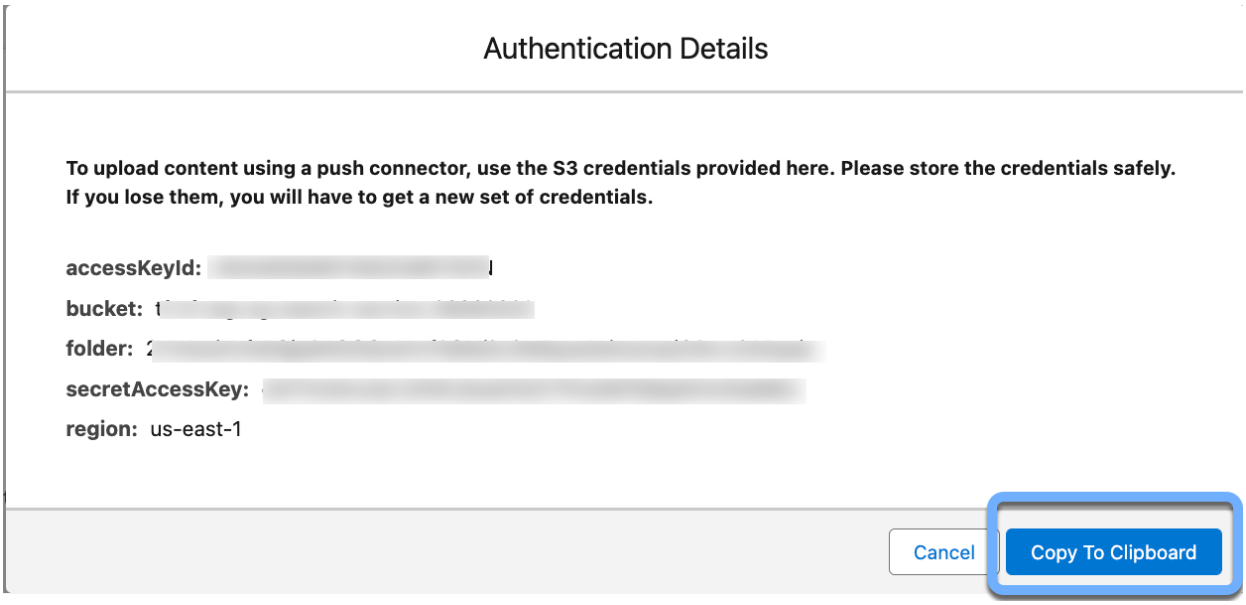
4. In the form that opens, fill in the fields. For specific information about what you should enter in each field, refer to the Source Reference table.



5. Click **Save**.



6. A pop-up appears with Authentication details. Click **Copy to Clipboard**. Save the information in a separate, secure location. You will need this information later. Do not lose it as you will not be able to retrieve it. If you lose these details, you will need to repeat the procedure from the beginning. **Important:** Safeguard the Authentication Details. Possession of these details allows the barer the ability to upload any file to Salesforce.



7. Using the information you copied from step 6, create a table like the one here. Fill in the values you collected from the pop-up screen in the third column. Save this information in a safe location, you will need it in the next step.

Received from Unified Knowledge As:	Name Value in GitHub	Value from Unified Knowledge
bucket:	AWS_BUCKET	
folder:	AWS_DIRNAME	
accessKeyId:	AWS_KEY_ID	
secretAccessKey:	AWS_SECRET_ACCESS_KEY	
region:	AWS_REGION	

Important: Safeguard the Authentication Details. Possession of these details allows the barer the ability to upload any file to Salesforce.

Next Steps

- Prepare Your GitHub Repository
- Syncing and Monitoring GitHub Markdown Content Uploads

Prepare Your GitHub Repository

This guide explains how to add a GitHub repository as a content source to Salesforce Unified Knowledge. Once you set up a GitHub repository as a Unified Knowledge source, any supported file located in the repository can be uploaded to Salesforce automatically.

Before you begin:

- If you are not the owner of the GitHub repository, contact your repository owner for permissions. Alternatively, you can ask them to perform the following procedure for you.
- Make sure you have a GitHub account and repository with markdown content.
- Make sure you have created the Unified Knowledge Source **first**.

Step 1 – Create GitHub Secrets for the Unified Knowledge Authentication Details

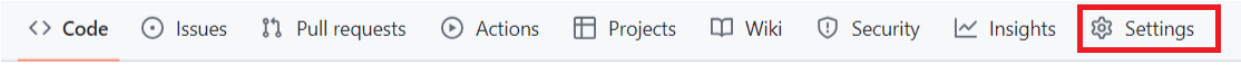
In this step you will take the authentication details you received from Unified Knowledge and create GitHub Secrets. If you are not the owner of the repository, you may need their assistance to perform this action.

Before you begin:

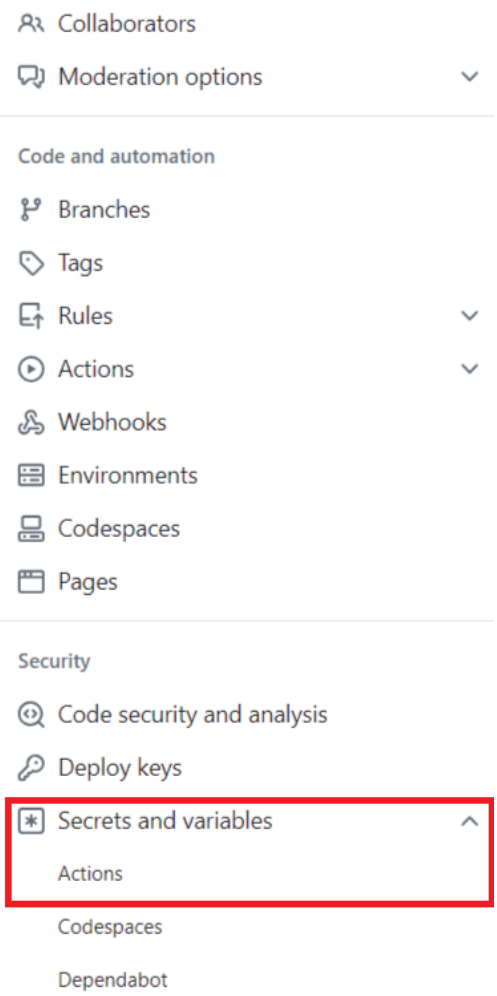
- Get the table you created in Configure Your GitHub Repository as a Unified Knowledge Source.

To create a GitHub Secret:

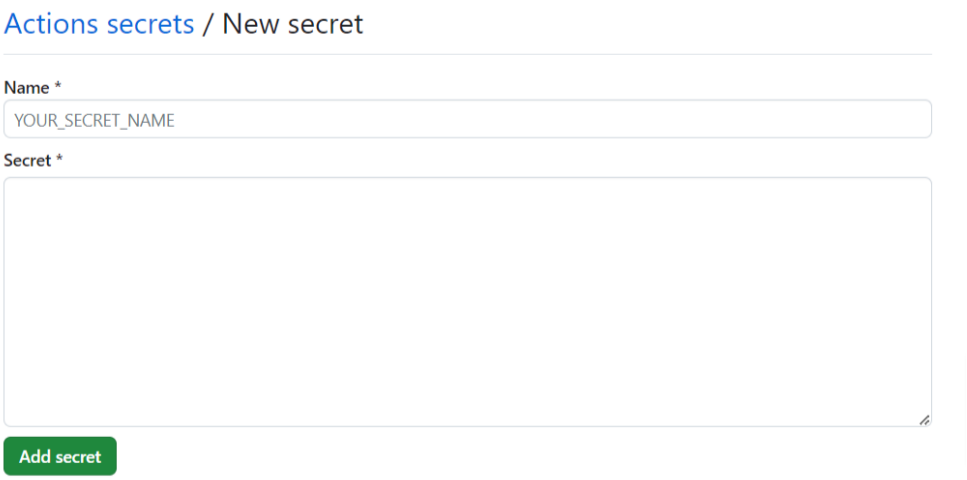
1. In Your GitHub repository click **Settings**



2. On the left menu expand **Secrets and variables**, and select **Actions.**

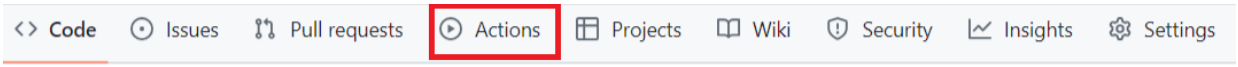


3. Click on the button – **New repository secret.**
4. In the screen that opens, fill secret name and value. Copy the name and values from the table you created in LINK
5. Click **Add secret.**
6. Repeat for all five secrets from the table you created in Step 7 on the previous page

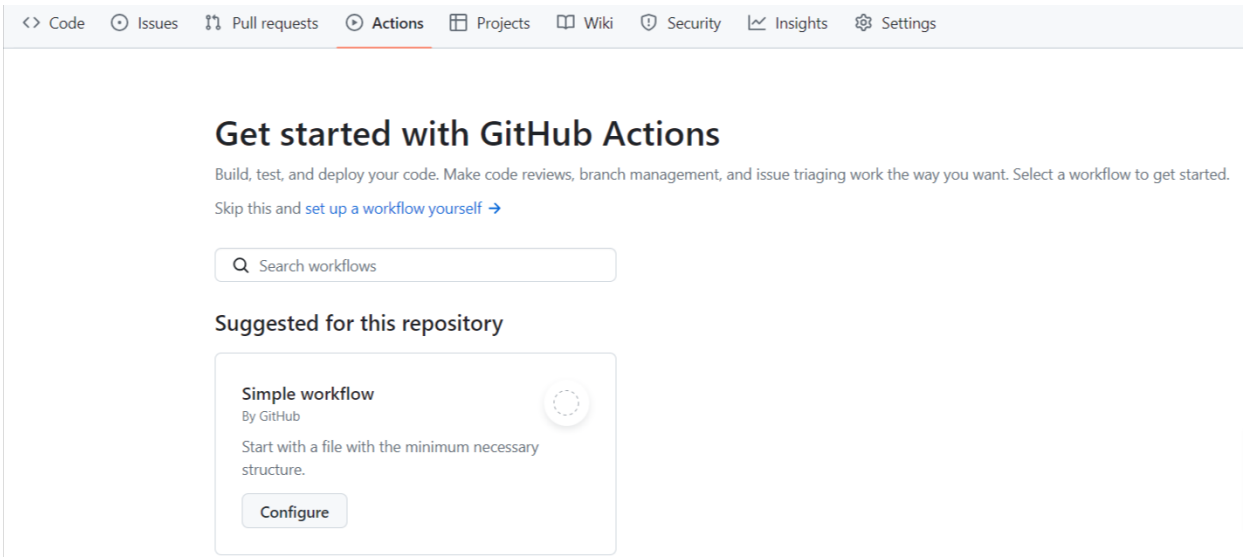


Step 2 – Create a GitHub Workflow

1. From the top menu bar, click **Actions.**



2. If no actions were configured in this repository, You will see something like this.



3. Click on “**set up a workflow yourself**” link.

4. In the page that opens paste one of the following snippets. The first one is for ad-hoc upload or manual upload. In this case you need to launch the action to upload the content. The second snippet is for automated upload. It currently runs when the Pull Request is closed but you can change this to your preference.

    1. **Manual Code Snippet**

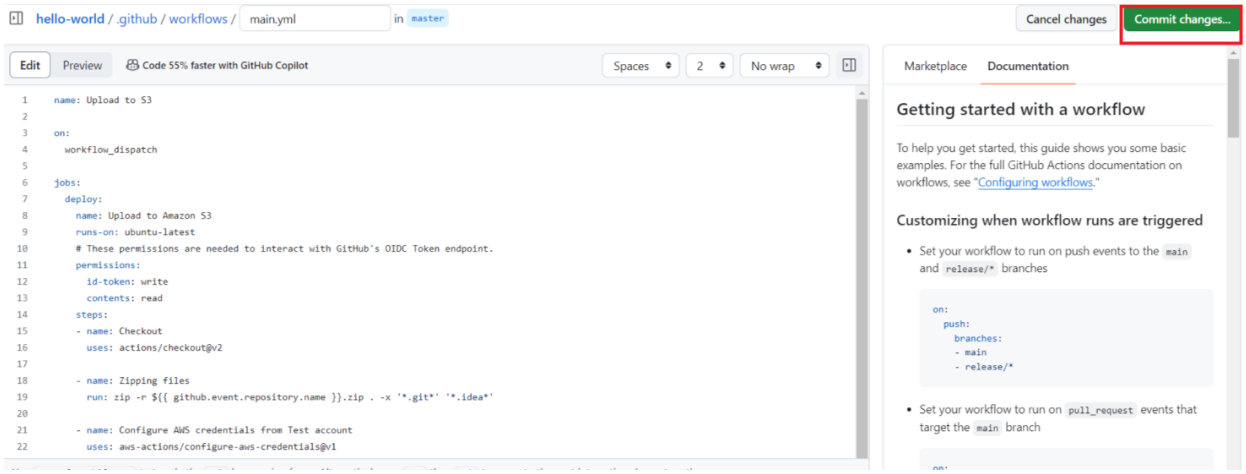
```
name: Manually Upload to Salesforce Unified Knowledge

on:
  workflow_dispatch

jobs:
  deploy:
    name: Upload to Amazon S3
    runs-on: ubuntu-latest
    # These permissions are needed to interact with GitHub's OIDC Tok
en endpoint.
    permissions:
      id-token: write
      contents: read
    steps:
    - name: Checkout
      uses: actions/checkout@v2

    - name: Zipping files
      run: zip -r ${{ github.event.repository.name }}.zip . -x '*.git
*' '*.idea*'

    - name: Configure AWS credentials from Test account
      uses: aws-actions/configure-aws-credentials@v1
      with:
        aws-access-key-id: ${{ secrets.AWS_KEY_ID }}
        aws-secret-access-key: ${{ secrets.AWS_SECRET_ACCESS_KEY }}
        aws-region: ${{ secrets.AWS_REGION }}

    - name: Copy files to the test website with the AWS CLI
      run: |
        aws s3 cp ${{ github.event.repository.name }}.zip s3://${{ se
crets.AWS_BUCKET }}/${{ secrets.AWS_DIRNAME }}/${{ github.event.repos
itory.name }}.zip
```

    2. **Automated Code Snippet**

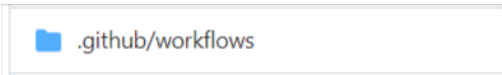
```
name: Automatically Upload to Salesforce Unified Knowledge

on:
  pull_request:
    branches: [main, master]
```

```
    types:
      - closed
  workflow_dispatch:

jobs:
  deploy:
    name: Upload to Amazon S3
    runs-on: ubuntu-latest
    # These permissions are needed to interact with GitHub's OIDC Tok
en endpoint.
    permissions:
      id-token: write
      contents: read
    steps:
    - name: Checkout
      uses: actions/checkout@v2

    - name: Zipping files
      run: zip -r ${{ github.event.repository.name }}.zip . -x '*.git
*' '*.idea*'

    - name: Configure AWS credentials from Test account
      uses: aws-actions/configure-aws-credentials@v1
      with:
        aws-access-key-id: ${{ secrets.AWS_KEY_ID }}
        aws-secret-access-key: ${{ secrets.AWS_SECRET_ACCESS_KEY }}
        aws-region: ${{ secrets.AWS_REGION }}

    - name: Copy files to the test website with the AWS CLI
      run: |
        aws s3 cp ${{ github.event.repository.name }}.zip s3://${{ se
crets.AWS_BUCKET }}/${{ secrets.AWS_DIRNAME }}/${{ github.event.repos
itory.name }}.zip
```

5. Click **Commit changes**.

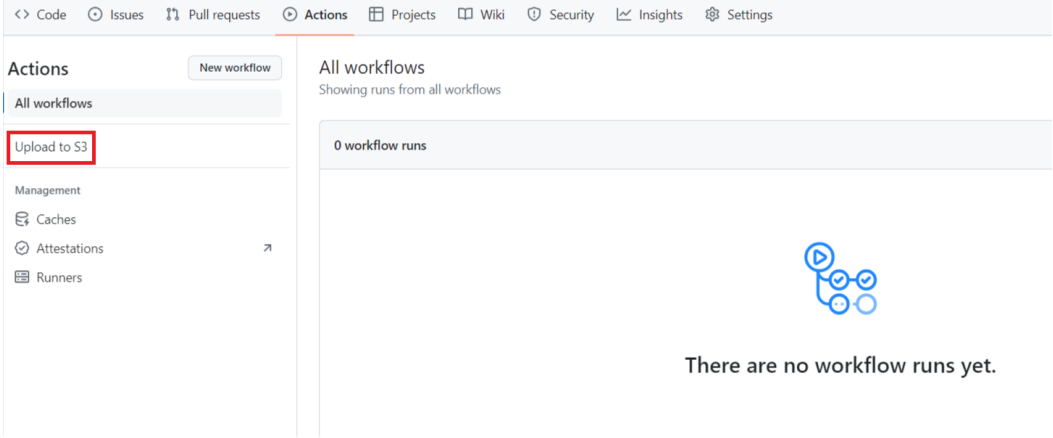


6. After committing changes, verify that your repository contains a **.github/workflows** folder with a **main.yml** file.

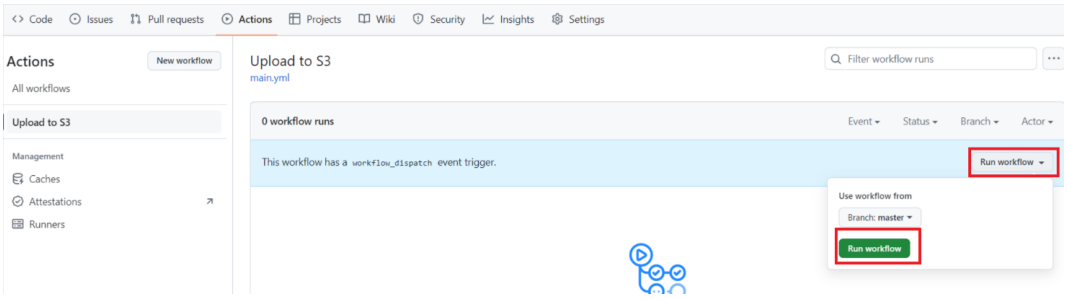


Step 3 – Run workflow

1. Go to **Actions**, select Upload to S3.

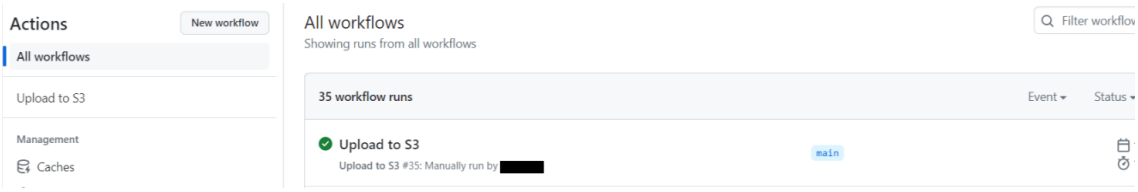


2. Click **Run workflow**

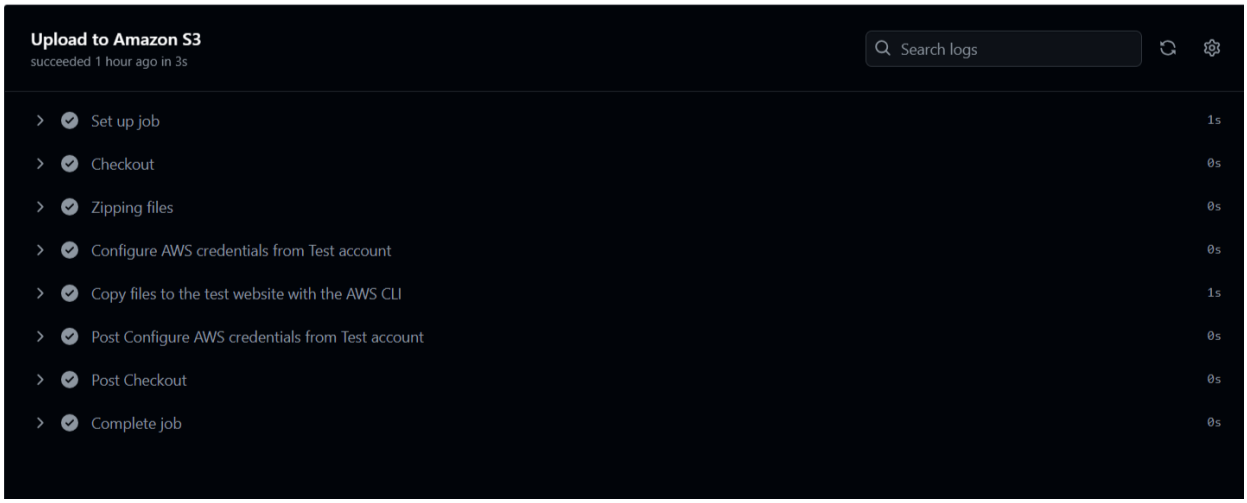


3. Refresh the actions page.

4. Confirm you see the workflow you created.



5. For more information about the publishing process, click the workflow name to expand. After the workflow is finished, confirm you see something similar to:



6. At this point your Connector starts uploading and you should see the connector status change in Unified Knowledge.

Next Step

Syncing and Monitoring GitHub Markdown Content Uploads

Syncing and Monitoring GitHub Markdown Content Uploads

Unified Knowledge Update Time Interval

Following the initial setup, your content will not upload until you run the workflow. At this point, Unified Knowledge uploads the content for the first time. The Initial upload time will vary depending on the size of your content. Once the initial upload is complete, your subsequent uploads run as per the trigger condition of your GitHub Workflow. If it is executed manually you will need to re-run the workflow each time you want to publish. If it is automatic, it will run each time a PR is merged.

Monitoring

Once you have configured a source for Unified Knowledge, you can see it, together with your other sources, in the Unified Knowledge Sources Dashboard. You can filter or sort the dashboard to make it easier to monitor a specific source.

The Unified Knowledge dashboard contains the following information:

- **Source**, **Label**, and **Developer name** are derived from the configuration settings for your Source.
- **Total Items per Connector** reflects the number of items uploaded from your source. Note that initially this number reflects all of your uploaded content, but after syncing, the number in this column reflects the number of items that were modified, and not your total content.
- Your connector **Status** as described in the table below.

The following table explains the status values you may see:

Status	Description
Unavailable	This is the initial status. It’s displayed when the source record was just created and the status of the article import job is not yet available.
Queued	The current import is in the queue, waiting to be processed.
Syncing	The article import job is in progress.
Completed	The article import job completed successfully.
Failed or Completed_with_errors	The article import failed to complete.
Invalid_Credentials	A connection to the external source couldn’t be established.
Ended	The article import job was canceled.
Timed_Out	The article import job took too long, or a connection couldn’t be established and the server timed out.
Zoomin Unavailable	Zoomin timed out, this can happen if the content size is larger than the size limitation.

Manage Your Source Configuration Settings

When you click the down arrow for any connector in the Unified Knowledge Source Dashboard, you can do the following actions:

- Force a full resync
- Delete a Source Connector
- Edit the Source Connector’s settings

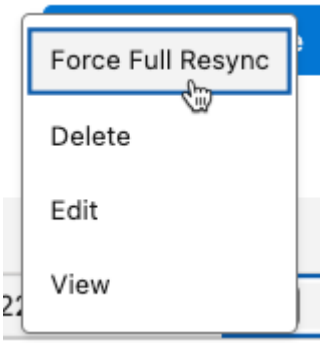
- View the Source Connector's configuration

Force a Full Resync

Forcing a full resync pushes all content for the specified connector from the Source to Unified Knowledge. If you changed your connector's configuration or want to bring in new material quickly, consider running this procedure.

To start a full resync:

1. In the Source Status Dashboard, click the down arrow to open the context menu for the connector you want to start the resync with.



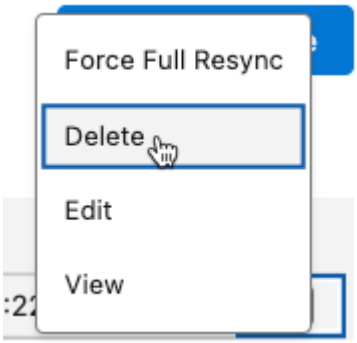
2. Click **Force Full Resync**.
3. You return to the monitoring dashboard and the connector's status changes to Syncing.

Delete a Source

Deleting a source severs the connection from Unified Knowledge to the Source and deletes all of the associated KB articles that were created. This action is permanent.

To delete a source:

1. In the Source Status Dashboard, click the down arrow to open the context menu for the connector you want to delete.



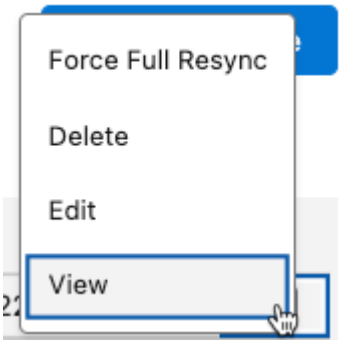
2. Click **Delete**.
3. You return to the monitoring dashboard and the source is removed from the table.

View a Source Configuration Setting

You can view your Source connectors settings and decide if you want to change anything. With this procedure, you cannot make changes, only view them.

To view a Source Connector's Configuration:

1. In the Source Status Dashboard, click the down arrow to open the context menu for the connector you want to view.



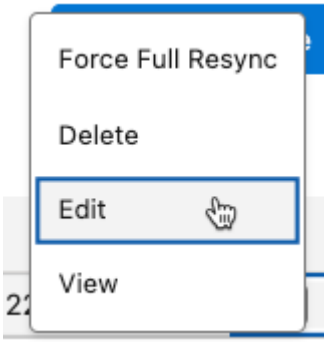
2. Click **View**. The View screen opens.
3. Click **Cancel** to return to the main screen.

Editing a Source's Configuration Settings

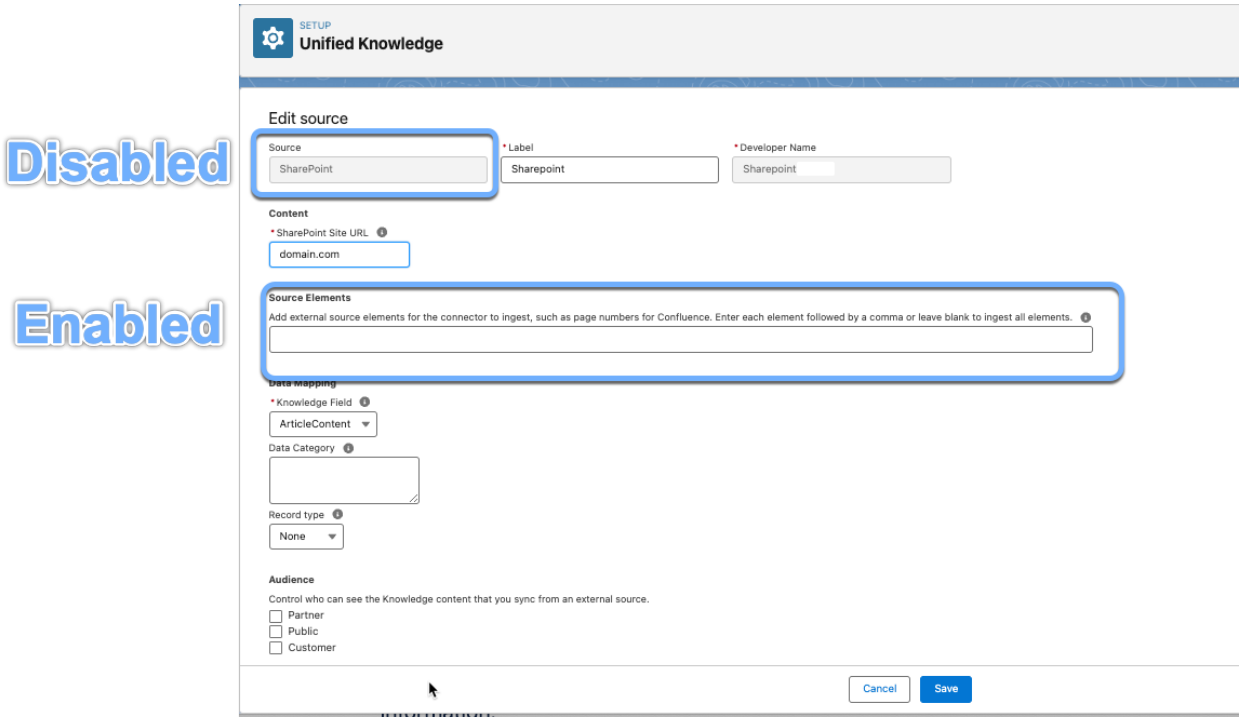
You can change some, but not all of your Source connectors settings. In the case where the change is not possible, you can delete the connector and create a new one.

To edit a Source Connector's Configuration:

1. In the Source Status Dashboard, click the down arrow to open the context menu for the connector you want to edit.



2. Click **Edit**. The Edit screen opens. If a field is disabled or missing, you cannot edit it. In all cases, you will not be able to change your authentication settings. This is why it is so important to test the connection before you save the initial configuration. If the Authentication settings are incorrect, you need to create a new source connector.



3. Change the parameters you want to change. Use the Source Reference Table for parameter information.

4. Click **Save**.