madcap software

USER GUIDE

MADCAP FLARE 2024

MadCap Central Integration

Copyright © 2024 MadCap Software. All rights reserved.

Information in this document is subject to change without notice. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of those agreements. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or any means electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of MadCap Software.

MadCap Software
9171 Towne Center Drive, Suite 335
San Diego, California 92122
858-320-0387
www.madcapsoftware.com

THIS PDF WAS CREATED USING MADCAP FLARE.

CONTENTS

CHAPTER 1

Introduction 6

CHAPTER 2

General Information for MadCap Central and Flare 9
- MadCap Central Window Pane 10
- Source Control and MadCap Central 12
- Moving From Dual-Bound to Single-Bound 32
- Central Analytics 35
- Private Outputs on MadCap Central 44
- Flare Features Not Supported in MadCap Central 63
- Security and MadCap Central 67
- MadCap Central Features Per Output Type 80

CHAPTER 3

Logging In and Out 81
- Logging In to Central 82
- Logging Out of Central 82
- The SSO Login Experience 83
- What's Noteworthy? 92

CHAPTER 4

Binding a Project to MadCap Central 93
Permission Required? 94
How to Upload (Bind) a Project to Central 95

CHAPTER 5

Importing Projects From Central 98
Permission Required? 100
How to Import a Project from Central 101

CHAPTER 6

Viewing Project Properties 102
How to View Project Properties in Flare 103

CHAPTER 7

Source Control Activities 104
Committing and Synchronizing (Pull, Push) in a Single-Bound Model 105
Pushing in a Dual-Bound Model 110

CHAPTER 8

Analytics Activities 112
Creating and Editing Analytics Keys 113
Associating Analytics Keys With Targets 114
Managing Analytics Keys 115
Downloading Analytics 116

CHAPTER 9

Central Review Process 117
Workflow 119

Benefits 130
What's Noteworthy? 131
Sending Files for Review on Central 132
Closing Central Review Files 139
Accepting or Rejecting Tracked Changes (Central Reviews) 142
Accepting Reviewed Files 146

CHAPTER 10

Publishing Directly to Central 151
Permission Required? 153
How to Publish Directly to Central 153
Command Line 155
What's Noteworthy? 157

CHAPTER 11

Removing Bindings 158
Permission Required? 159
How to Remove a Binding from a Project in Central 159

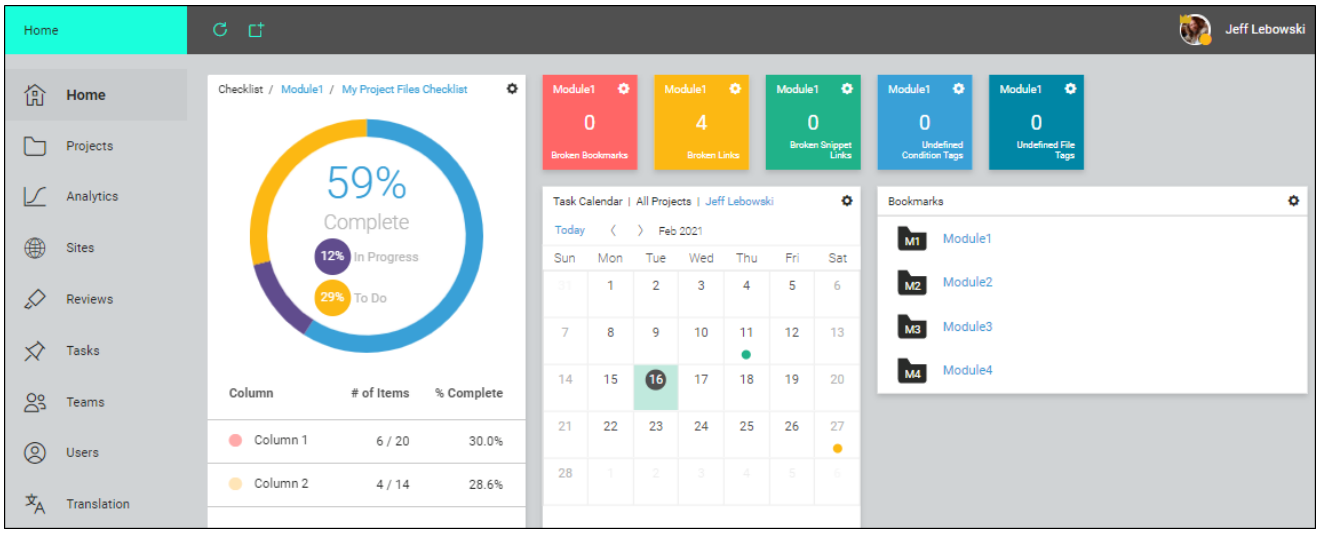
APPENDIX

PDFs 160
Tutorials 160
Cheat Sheets 161
User Guides 162

CHAPTER 1

Introduction

MadCap Central is a cloud-based platform that lets you plan, track, and manage the processes, content, and teams that are at the heart of your organization.



MadCap Central's integration with MadCap Flare lets you store copies of your projects in Central, continue to work on them locally in Flare, and keep both sets of copies in sync. You can use Central to quickly build and publish output (and roll back when necessary) without the need to involve an IT department. Custom vanity URLs let you produce meaningful paths for your outputs. You can also send topics and snippets for review on Central, as well as use custom checklists to track your progress in Flare projects.

For more complete information about Central, see its online Help:

https://help.madcapsoftware.com/central

Benefits

- Configure single sign-on (SSO)
- Host output with custom domains and vanities
- Publish private output
- View analytics on output
- Configure for single sign-on
- Assign user permissions
- Use integrated checklists
- Collaborate with reviewers
- Create and manage tasks
- Integrate with Slack
- Request translation quotes
- Customize dashboards and widgets

General Information

- "MadCap Central Window Pane" on page 10
- " Source Control and MadCap Central" on page 12
- "Central Analytics" on page 35
- "Private Outputs on MadCap Central" on page 44
- "Flare Features Not Supported in MadCap Central" on page 63
- "Security and MadCap Central" on page 67
- "MadCap Central Features Per Output Type" on page 80

Main Activities

- "Logging In and Out" on page 81
- "Binding a Project to MadCap Central" on page 93
- "Importing Projects From Central" on page 98
- "Viewing Project Properties" on page 102

- "Source Control Activities" on page 104
- "Analytics Activities" on page 112
- "Sending Files for Review on Central" on page 132
- "Publishing Directly to Central" on page 151
- "Removing Bindings" on page 158

NOTE At this time, you cannot perform source control tasks from the Central interface. This can be done only from the Flare project.

NOTE MadCap Central is sold separately from Flare. Please contact MadCap Software Sales for more information.

NOTE Another product option that MadCap Software offers is Xyleme Syndicate, which is a cloud-based content delivery platform. You can create a Syndicate publishing destination and automate the publishing process to upload documents (e.g., eLearning, video, PDF, HTML) from Flare to a Syndicate folder.

CHAPTER 2

General Information for MadCap Central and Flare

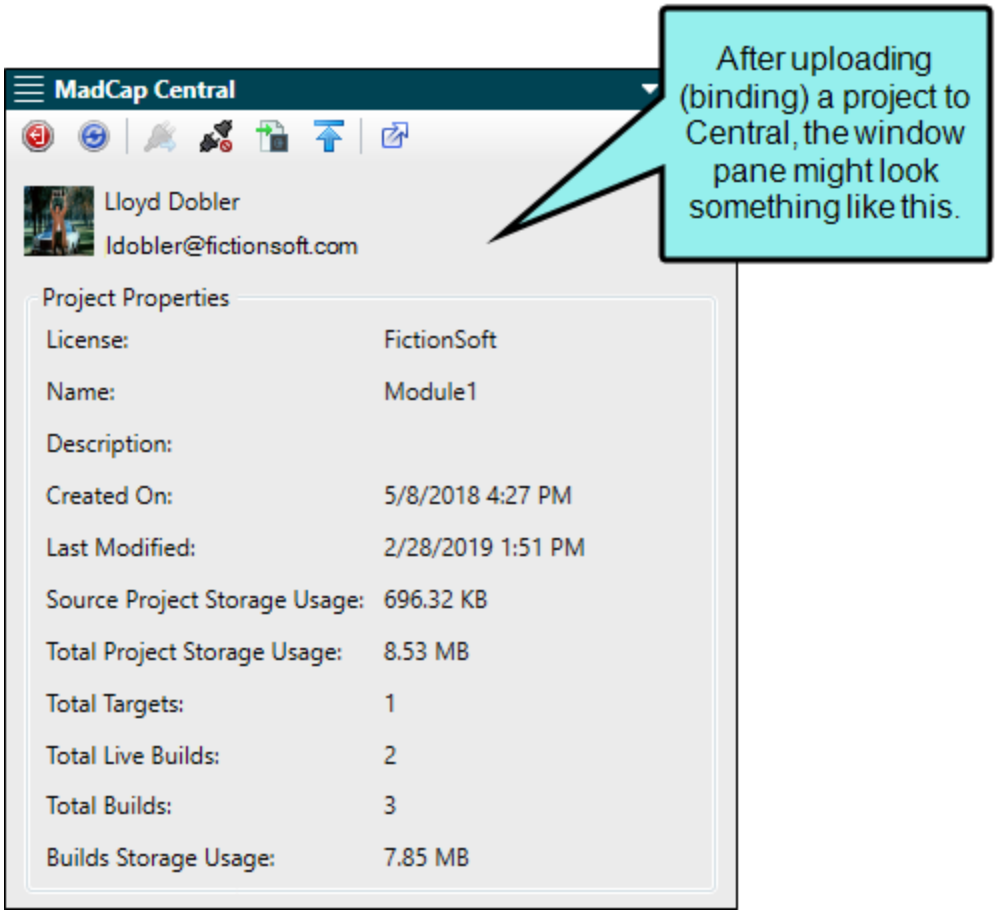
There are various pieces of general information you should know if you plan to use this feature.

This chapter discusses the following:

MadCap Central Window Pane 10
Source Control and MadCap Central 12
Moving From Dual-Bound to Single-Bound 32
Central Analytics 35
Private Outputs on MadCap Central 44
Flare Features Not Supported in MadCap Central 63
Security and MadCap Central 67
MadCap Central Features Per Output Type 80

MadCap Central Window Pane

The MadCap Central window pane in Flare lets you upload (bind) and import projects, as well as push project changes to Central. Additionally, you can see project properties, log in and out of your Central account, and launch the Central portal in your browser.



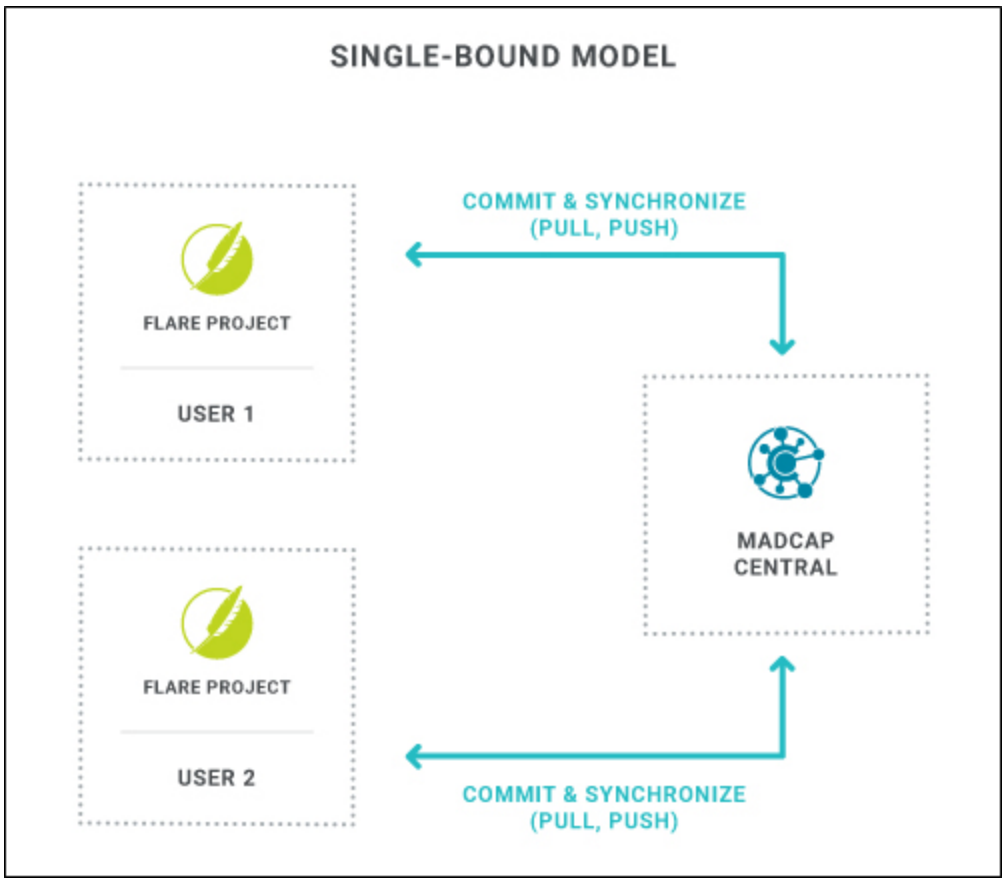
NOTE If the Central license is configured for single sign-on (SSO), you can log in through your company's identity provider. Otherwise, you would use the same password that you set in Central. For details on setting up SSO, see the Central online Help.

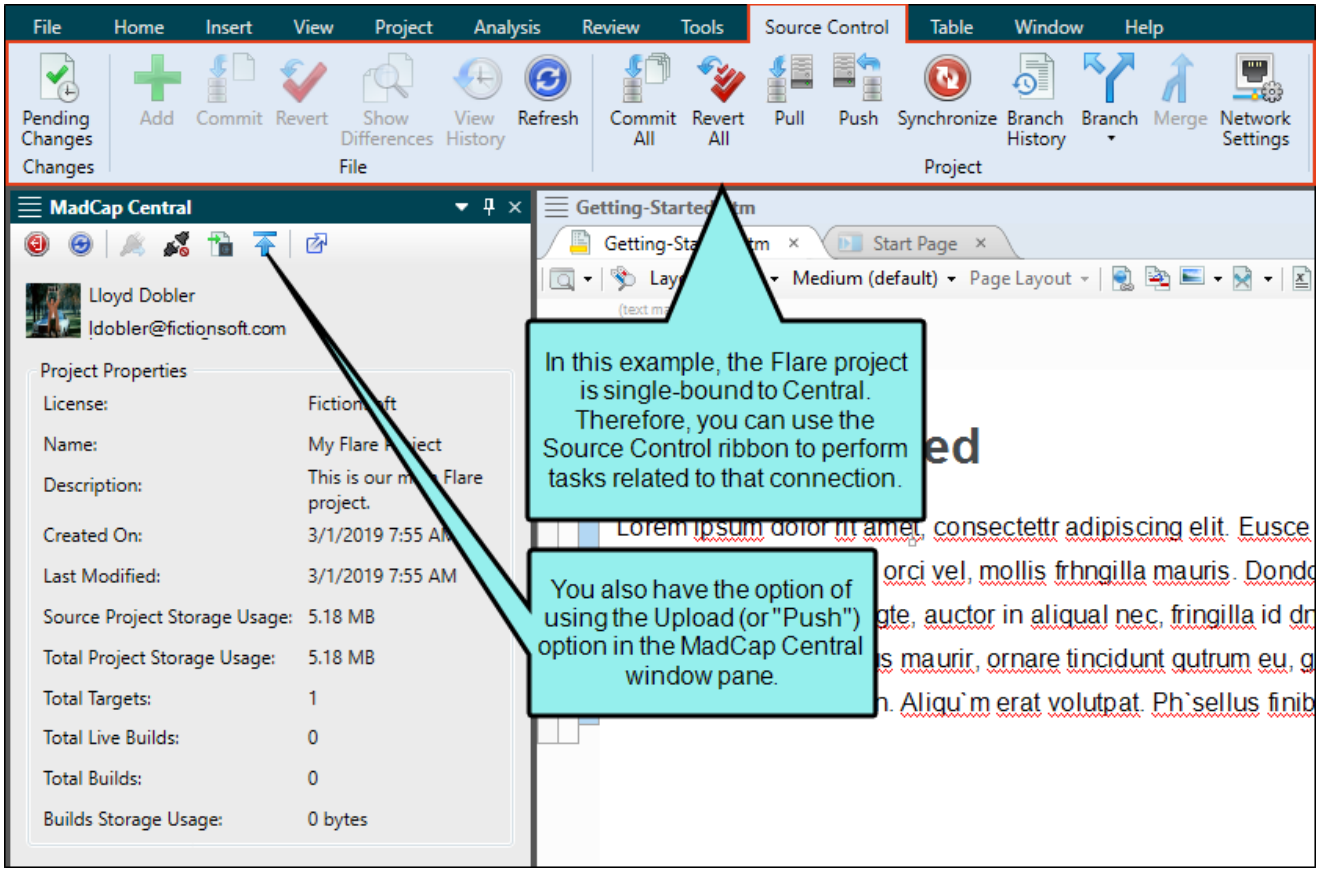
Source Control and MadCap Central

When you upload a Flare project to Central, the files are connected to Central via an integrated source control system (Git). Your interaction with source control can follow one of two models—single-bound (recommended) or dual-bound.

Single-Bound Projects

If your Flare project is not already bound to a third-party source control tool and you upload the project to Central, you will have only one source control binding. Therefore, you can use the source control connection between your local Flare project repository and the remote Central repository as your primary source control solution. With this model, you can use the Source Control ribbon in Flare to perform regular source control tasks such as committing and synchronizing (i.e., pulling and pushing) files. However, you also have the option of using tools outside of Flare (e.g., Git Bash) to perform source control tasks (e.g., in case there are certain source control tasks not supported in the Flare interface that you need to complete).


SINGLE-BOUND MODEL
COMMIT & SYNCHRONIZE
(PULL, PUSH)
FLARE PROJECT
USER 1
MADCAP
CENTRAL
FLARE PROJECT
USER 2
COMMIT & SYNCHRONIZE
(PULL, PUSH)


File
Home
Insert
View
Project
Analysis
Review
Tools
Source Control
Table
Window
Help
Pending Changes
Changes
Add
Commit
Revert
Show Differences
View History
Refresh
File
Commit All
Revert All
Pull
Push
Synchronize
Branch History
Branch
Merge
Network Settings
Project
MadCap Central
Lloyd Dobler
ldobler@fictionsoft.com
Project Properties
License:
Name:
Description:
This is our m Flare project.
Created On:
3/1/2019 7:55 AM
Last Modified:
3/1/2019 7:55 AM
Source Project Storage Usage:
5.18 MB
Total Project Storage Usage:
5.18 MB
Total Targets:
1
Total Live Builds:
0
Total Builds:
0
Builds Storage Usage:
0 bytes
Start Page
Medium (default)
Page Layout
In this example, the Flare project is single-bound to Central. Therefore, you can use the Source Control ribbon to perform tasks related to that connection.
You also have the option of using the Upload (or "Push") option in the MadCap Central window pane.

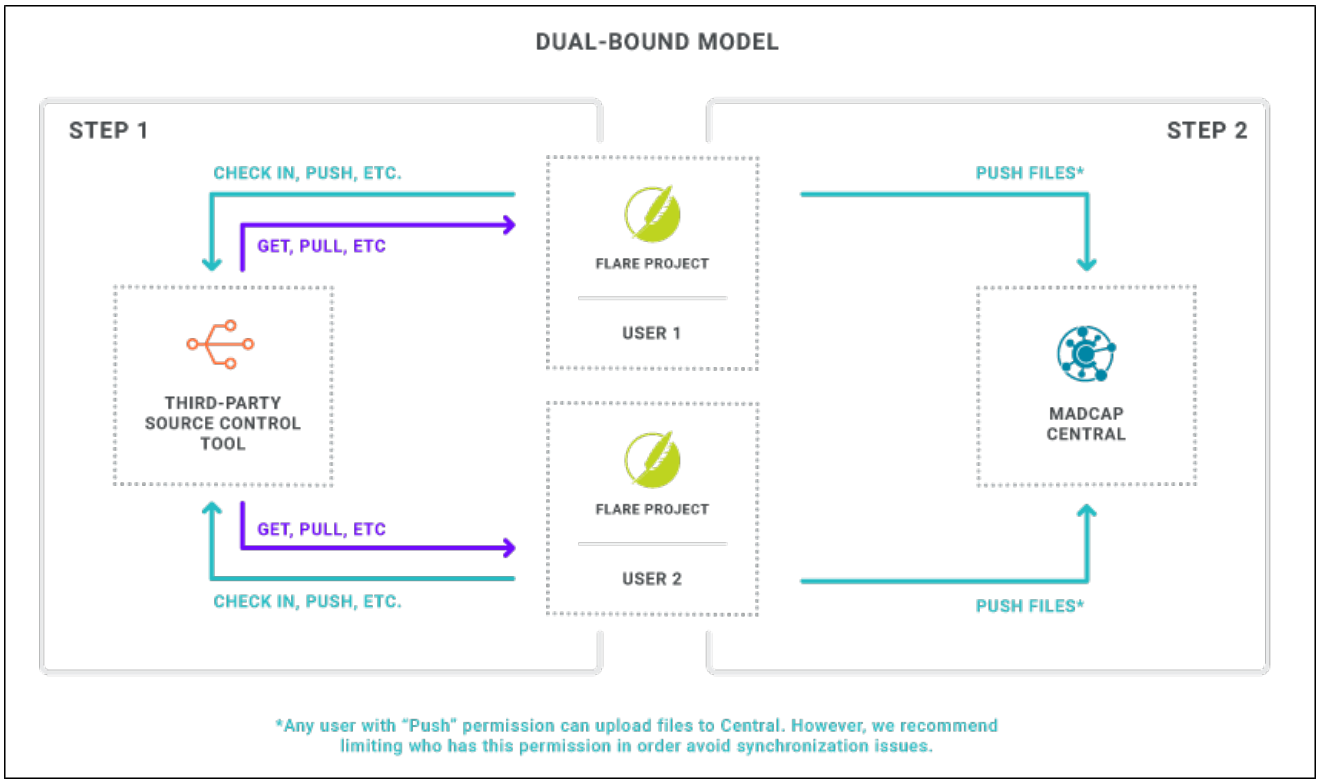
How to Use Single-Binding

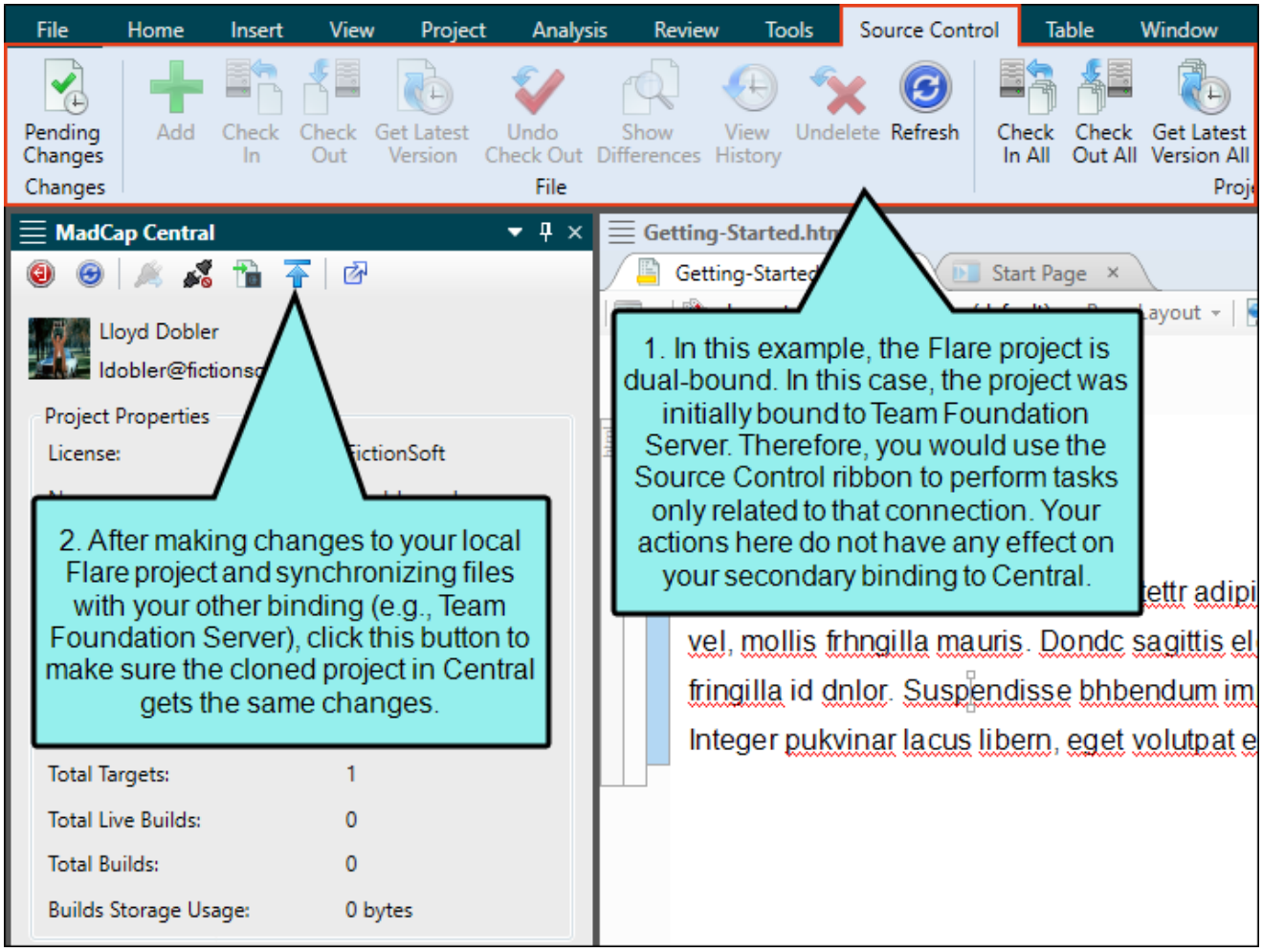
1. Open the project in MadCap Flare.
2. Select View > MadCap Central. The MadCap Central window pane opens.
3. Be sure to log in if you aren't already.
4. Upload (bind) your project to Central. See "Binding a Project to MadCap Central" on page 93.
5. In Central, assign users to the new project and make sure they have “Import/Pull” and “Push” permissions. This enables users to import the project using the MadCap Central window pane. It also allows users to push changes up to Central. See the Central online Help.
6. Newly assigned users must now import the project from Central using the MadCap Central window pane in Flare. This allows each user to work on the project locally. See "Importing Projects From Central" on page 98.
7. In Flare, make changes to your files. Then commit those files and synchronize them (i.e., pull, then push) with the cloned project on Central. See "Committing and Synchronizing (Pull, Push) in a Single-Bound Model" on page 105.

Dual-Bound Projects

A dual-bound project has a first binding to a third-party source control provider—such as Git, Perforce Helix Core, Subversion, or Team Foundation Server—and a second binding to Central. Some might choose this model because it allows them to use a source control provider they're comfortable with, while also taking advantage of features on Central.

However, if possible, it's recommended that you use the single-bound setup since it's simpler and more streamlined. With the dual-bound model, you use the first source control binding to do most of the version control work; the Source Control ribbon in Flare is used only for tasks related to the first binding. Then periodically, you will use the MadCap Central window pane in Flare to upload (or "push") the latest files from your local copy of the project up to Central via the second binding; the Source Control ribbon in Flare is not used for the second binding. However, keep in mind that creating and synchronizing branches (other than master) between Flare and Central is supported only with a single-bound project or with a Git-Central dual-bound setup.





NOTE It is possible for multiple people working on a dual-bound project to push files to Central. However, if you are using a source control provider other than Git for the first binding, the most recently pushed files are the ones that are used in Central. In other words, the last person to push "wins." To avoid issues, be sure that you have the most recent version of file changes from all other writers in your local project before you push. You may even want to limit users' permissions (see the Central online Help) so only one or two people are allowed to push files to Central.

NOTE Your first source control binding (between Flare and a third-party provider) should be done from the Flare interface, rather than from another tool. Otherwise, Flare and Central will not recognize that binding. If you already have a project that was bound using another tool, you have a couple of options:

- You can remove the binding and then bind again using Flare.
- You can create a new Flare project, importing from source control. This method allows you to retain the repository.

NOTE If you are using Git as your third-party source control tool for the first binding, the dual-bound model works slightly differently than it does if you are using another source control provider. In this situation, you will still use the MadCap Central window pane and Source Control ribbon in Flare to manage your changes. However, you are able to pull and push files from and to *either* Central or your Git repository. As a result, your Git repository and your Central repository might be completely different, and you may encounter conflicts. In this situation, Flare's Conflict Resolution dialog will open and you can accept or reject the changes. It is recommended that you establish an internal workflow to dictate the order in which you pull and push from and to each repository to prevent conflicts and ensure that your files in Central stay up-to-date. However, an even better solution is to use a single-bound setup instead of dual-binding.

How to Use Dual-Binding—Perforce, Subversion, TFS, and Other Providers

1. Open the project in MadCap Flare, and make sure your project is already bound to a third-party source control provider. See the Flare online Help.
2. Select View > MadCap Central. The MadCap Central window pane opens.
3. Be sure to log in if you aren't already.
4. Upload (bind) your project to Central. See "Binding a Project to MadCap Central" on page 93.

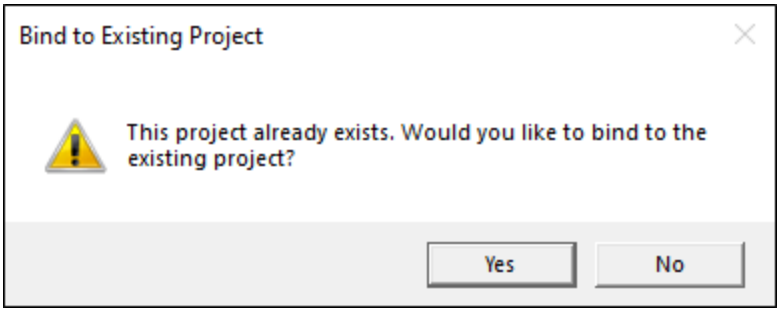
   Because you are dual-bound, Flare will prompt you to check out your Flare project file before you can upload the project.
5. Check in the Flare project file to your third-party source control provider. See the Flare online Help.
6. In Central, assign users to the new project and make sure they have “Push” permissions. This gives the users the ability to push local content to the project on Central. See the Central online Help.
7. Newly assigned users must now do one of the following, depending on whether or not they already have a copy of the Flare project on their local machine:
   - Have a Local Copy Get the latest version of the Flare project file from the third-party source control provider. See the Flare online Help.
   - Do Not Have a Local Copy Import the project from the third-party source control provider using the Import Project from Source Control Wizard. See the Flare online Help.

> NOTE If your project is already bound to a third-party source control provider other than Git—i.e., you are working in a dual-bound model as opposed to a single-bound model—the Import option in the MadCap Central window pane will be disabled. You will have to import your project directly from source control using your non-Central binding.
>
> If you already have the project on your machine, you do not need to reimport the project from source control. Instead, you can get the latest version from source control by using the Flare Source Control ribbon (see the Flare online Help).
>
> Be sure that you have the latest version of the project.

Either option will give users the most current version of the project.

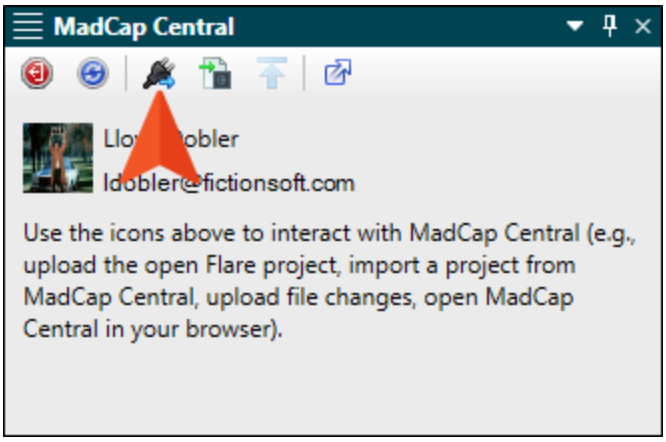
8. In Flare, make changes to your files. You should manage all of your file changes using your third-party source control provider, using the following actions:
   - Check In Check in your changes to source control. See the Flare online Help.
   - Get Latest Get the latest version of your teammates' changes from source control and add them to your local project. See the Flare online Help.
9. When you are done making changes, push your final changes to Central. To do this, click in the MadCap Central window pane. See "Pushing in a Dual-Bound Model" on page 110.

NOTE When a project is dual bound to Central and a non-Git third-party provider, keep the following in mind if you (i.e., the second user) want to access this setup. After importing the project from Perforce Helix Core, Subversion, or Team Foundation Server, you will have to re-bind the project to Central. In Flare's MadCap Central window pane, click to upload (or bind) the project to Central. In the Bind Project dialog, it is important to enter the exact name of the project as it currently exists on Central. When you click **OK**, a message displays asking if you want to bind to the existing project. Select **Yes**. This re-establishes the existing Flare/Central connection; note that it does not create a new binding.

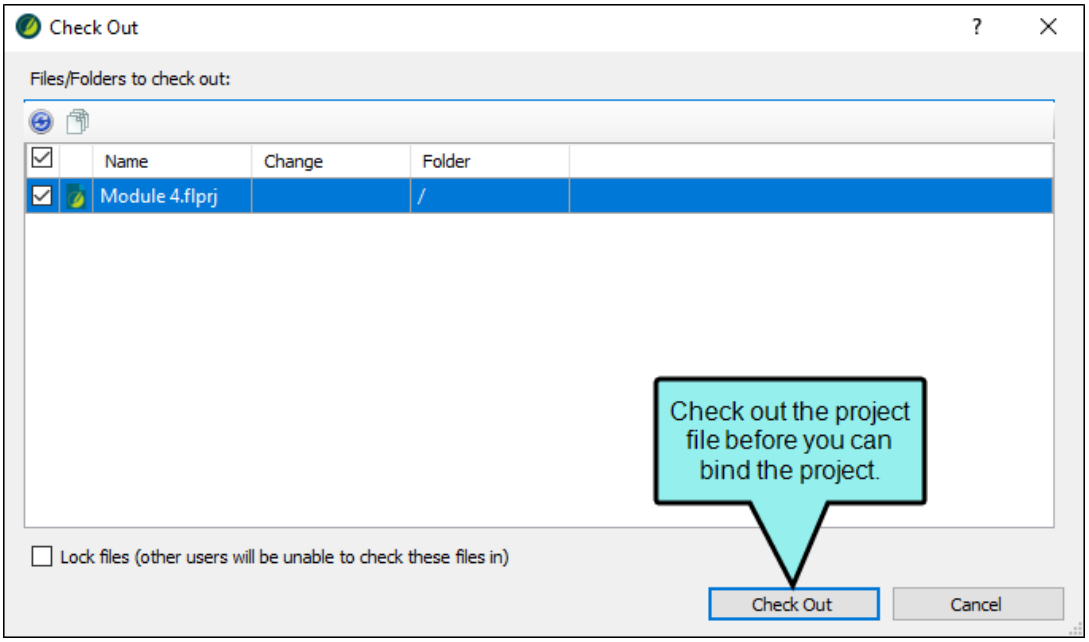


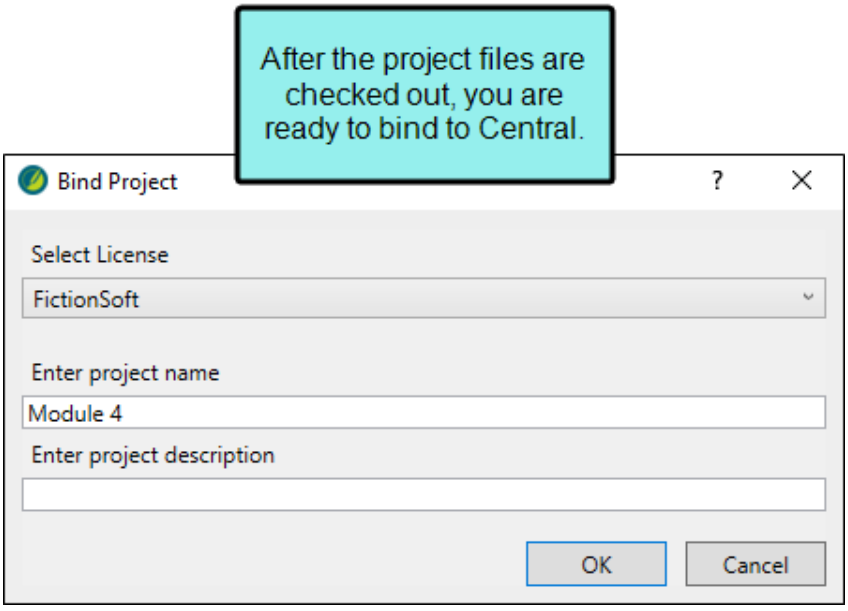
☆ EXAMPLE You are working on a team of writers and your project is bound to Microsoft Team Foundation Server (TFS). Therefore, that is your primary source control provider, and the source control connection between Flare and Central serves as a secondary source control provider.

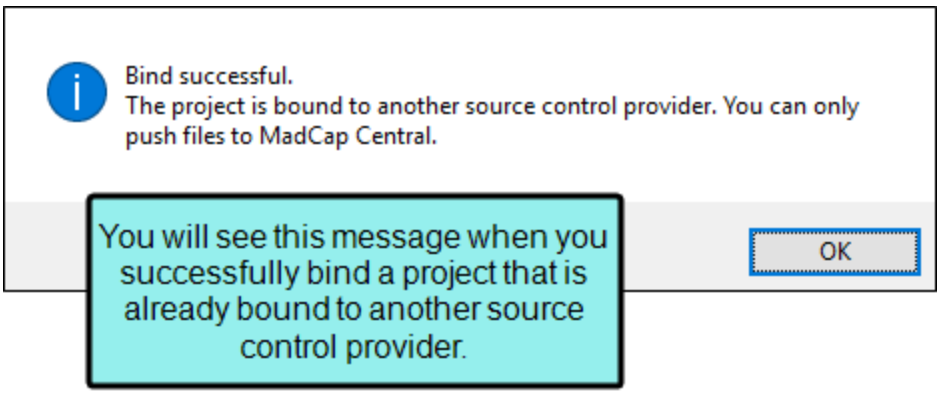
When you first start working with Central, you have to upload (or bind) the project to Central. To do this, click in the MadCap Central window pane.

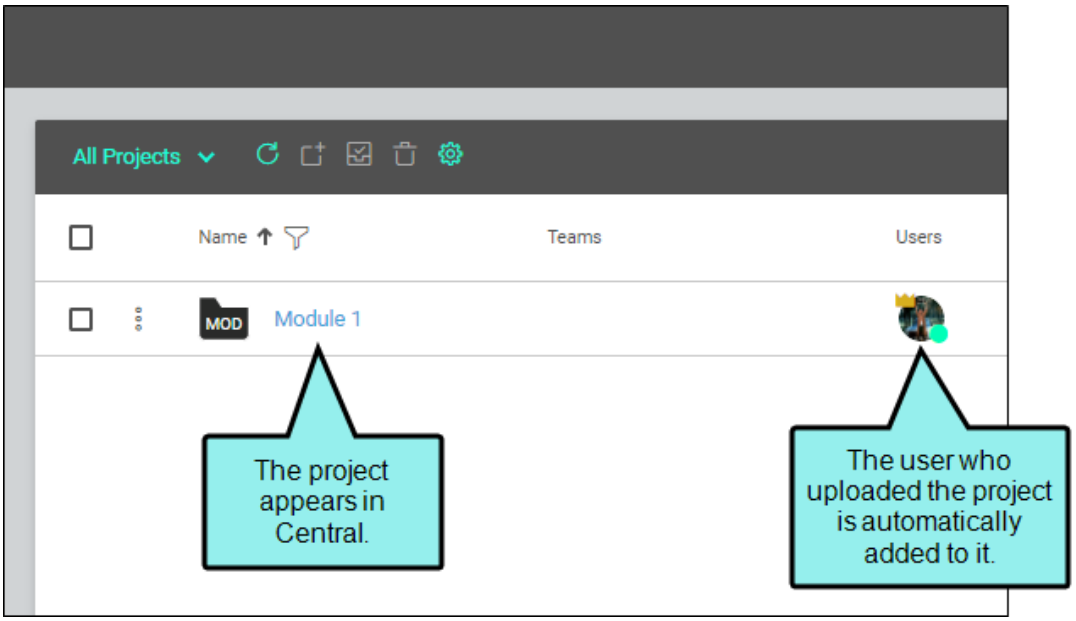


☆ Because the project is bound to TFS, you will have to check out the project file before you can bind the project (Flare will prompt you to do this). Binding the project will establish the connection between Flare and Central, and creates a copy of the project in Central.

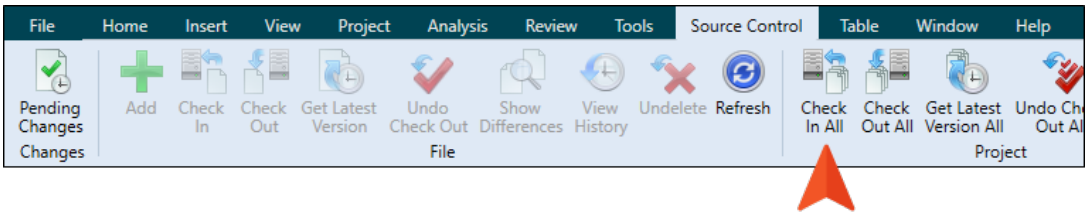





Bind successful.
The project is bound to another source control provider. You can only push files to MadCap Central.
OK
You will see this message when you successfully bind a project that is already bound to another source control provider.

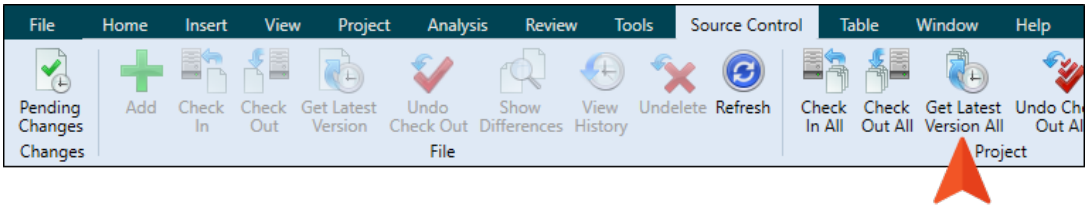

All Projects
Name
Teams
Users
MOD
Module 1
The project appears in Central.
The user who uploaded the project is automatically added to it.

☆ Click **Check In All** in the Flare Source Control ribbon.

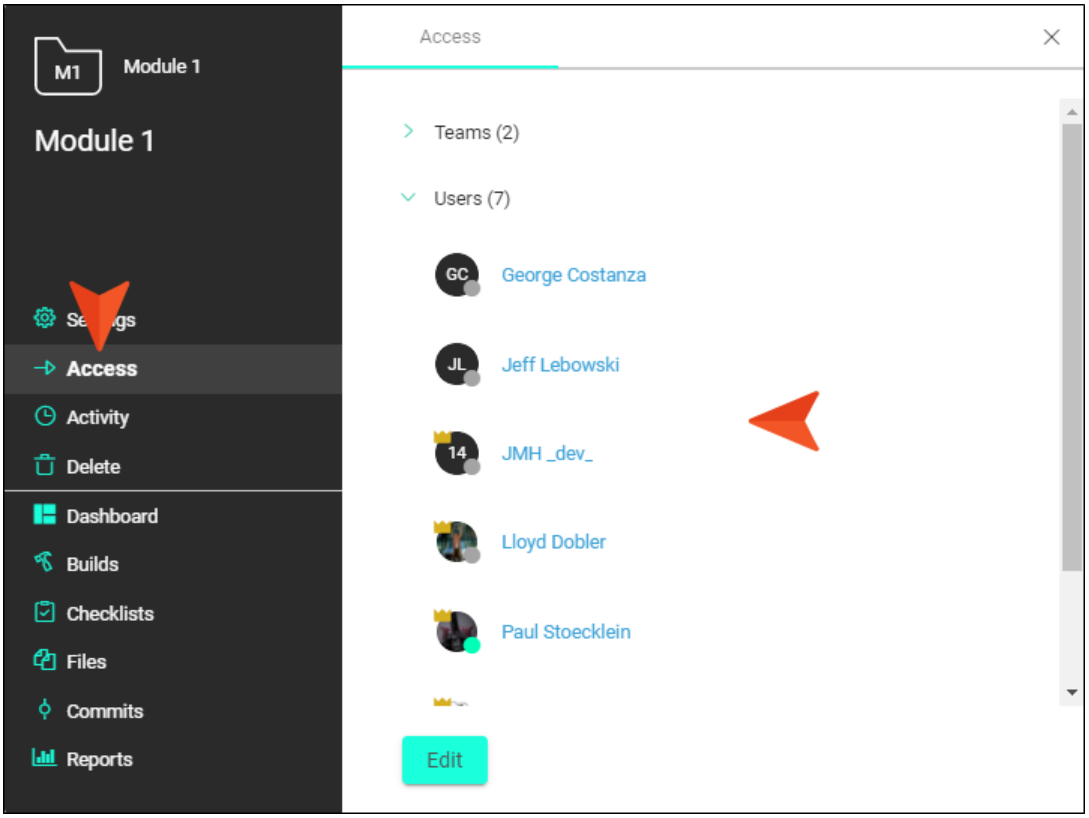


If any writers on the team do not already have a local copy of the project, they should import the project from TFS so they can work on the project locally (they cannot import from Central because the project's primary source control location is TFS). To import the project, they will use the Import Project From Source Control Wizard. For more information about importing projects from source control, see the Flare online Help.

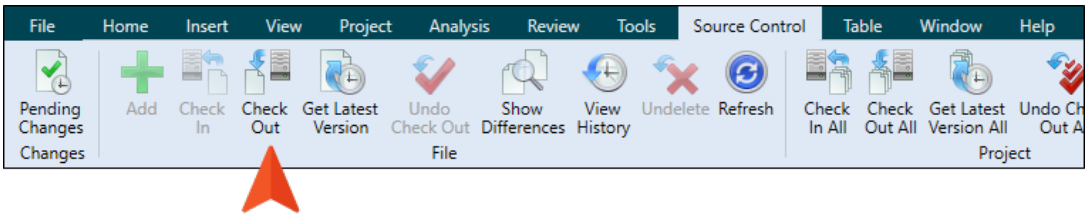
Writers on your team who already have the project locally do not need to reimport. They simply need to get the latest version of the project by clicking **Get Latest All** in the Flare Source Control ribbon.



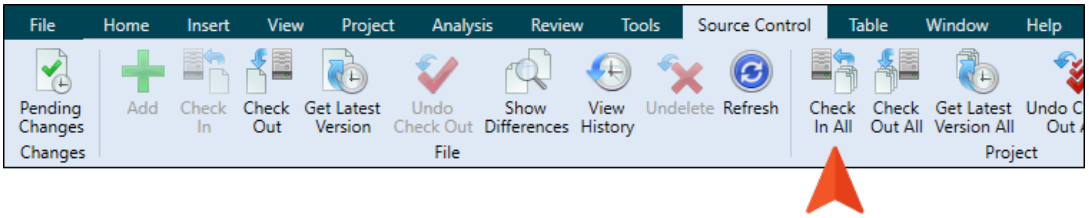
Now that everyone has the updated project file, you can all work on the project and make changes using TFS. However, a user with the "Manage Teams/Projects" permission also needs to assign your teammates to the project in Central (see the Central online Help).



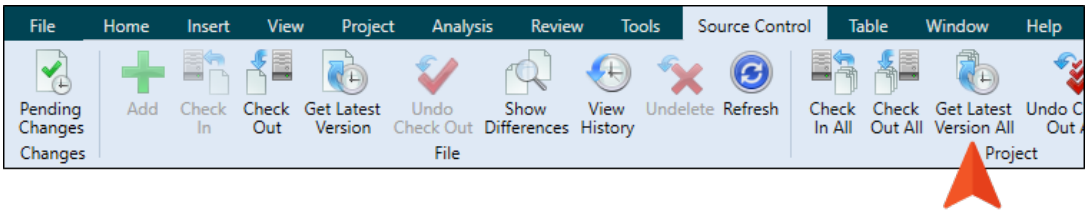
As you work on files in Flare, you should check out your files using Flare's **Check Out** feature on the Flare Source Control ribbon (or using automatic checkouts, if you have this enabled).



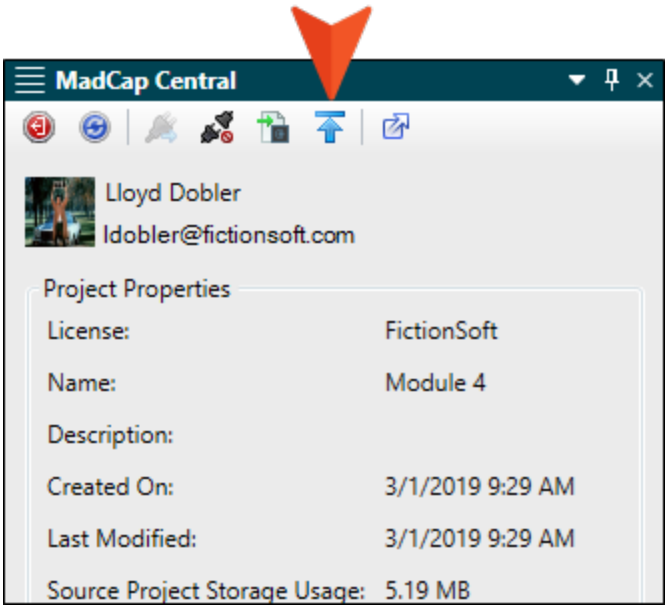
After you are finished making changes, you should check in your changes to TFS. You can do this by clicking **Check In All** on the Flare Source Control ribbon. This will upload your changes to TFS.



Once your changes are checked in, you should make sure that you have the latest version of your teammates' files, as well. To do this, click **Get Latest Version All** on the Flare Source Control ribbon. This will download the latest changes of all of the files to your local version of the project.



When you and your teammates are finished making changes, you need to synchronize your local copy of the project with the copy of the project in Central. To do this, click in the MadCap Central window pane.



This will push your local changes up to Central. It is important to remember that the last person who pushes their changes to Central wins, so be sure that everyone has the latest version of the project files before you push (or even limit some users' permissions so they are not able to push; see the Central online Help) so you do not have out-of-date files in Central.

Once your files are up in Central, you can publish output and manage your project.

How to Use Dual-Binding—Git

1. Open the project in MadCap Flare.
2. Be sure your project is already bound to Git. See the Flare online Help.
3. Make sure all changes are committed before binding to Central. See the Flare online Help.
4. Select View > MadCap Central. The MadCap Central window pane opens.
5. Be sure to log in if you aren't already.
6. Upload (bind) your project to Central. See "Binding a Project to MadCap Central" on page 93.
7. Push the Flare project file to Git. See the Flare online Help.
8. In Central, assign users to the new project and make sure they have “Push” permissions. This gives the users the ability to push local content to the project on Central. See the Central online Help.
9. Newly assigned users must now do one of the following, depending on whether or not they already have a copy of the Flare project on their local machine:
    - Have a Local Copy Pull the latest version of the Flare project file from Git. See the Flare online Help.
    - Do Not Have a Local Copy Import the project from Git using the Import Project from Source Control Wizard. See the Flare online Help.

      > IMPORTANT If you are using a Git-Central dual-binding, you can import projects from Git (using the Import Project From Source Control Wizard) or from Central. The project will be the same. However, if you import from Central, you will only be able to push changes to and pull changes from Central. If you import from Git, you will be able to push changes to and pull changes from Central *as well as* your main Git repository. This is because the project in Central has no connection to the original Git repository, and if you import from Central you will not have those source control bindings.

    Either option will give users the most current version of the project.

10. In Flare, make changes to your files. When working in a Git-Central dual-bound situation, you can make changes to both Central and Git, using the following actions:
    - **Pull** Pull your teammates' changes from Git and add them to your local project. See the Flare online Help.
    - **Push** Push your changes to source control. See the Flare online Help.

    NOTE It does not matter which location (i.e., Git or Central) you push to or pull from first, as long as you push the files to both Git and Central and that everyone on your team uses the same workflow.

11. (Optional) If the changes in Git or Central become out-of-sync with each other (i.e., changes are made in one location but not the other), the Resolve Conflicts dialog opens. If this happens, use the dialog to accept or reject other users' changes. See the Flare online Help.
12. When you are done making changes, push your final changes to Central. To do this, click in the MadCap Central window pane. See "Pushing in a Dual-Bound Model" on page 110.

IMPORTANT There are several different workflows you can use when working with a Git-Central dual-binding. The steps above are only one example of how you might perform this process. Because of the potential for file conflicts, it is essential that you establish a workflow for using a Git-Central dual-binding, and make sure that all your team members follow the same steps.

What's Noteworthy?

NOTE If you are using a dual-bound setup where you are bound to MadCap Central and a third-party provider, you might decide at some point to move to a single-bound configuration, removing one of the bindings but leaving the other.

NOTE If your team is using Git branching, you can create branches in Flare and push them up to Central. Other writers on your team who need to work in the same branches can pull them down from Central.

NOTE If you are using a Git-Central dual-binding, and you encounter conflicts in one repository (i.e., Git or Central), you will likely encounter them in the other location as well. Conflicts in one repository will most likely need to be resolved in the other repository. See the Flare online Help.

Moving From Dual-Bound to Single-Bound

If you have been using a dual-bound setup for source control, you might find that you want to change to a single-bound project. One reason you might decide to do this is branching, which previously was not available in MadCap Central, but is now supported for Git.

The steps for moving from a Git-Central dual-bound setup to a single-bound setup are slightly different than those for other provider configurations (i.e., Perforce Helix Core, Subversion, Team Foundation Server). Keep in mind that if you are using a branching solution with a non-Git providers, those branches are still not be supported in Central. You can only push branches to Central if you have a single-bound setup or a Git-Central dual-bound setup.

How to Move From Dual-Bound to Single-Bound for a Git-Central Setup

If your first binding is to Git and your second is to Central, complete the following steps in your Flare project.

1. Each author on the team should commit all outstanding changes and synchronize each local branch with the corresponding remote branch.
2. It's possible, or even likely, that not all writers on the team have all of the remote branches locally, and they don't necessarily need to. But between all of the team members, all of the remote branches that you want to keep should be pulled down to someone's local repository by selecting the remote branches and switching to them via Flare's Branch Management dialog. In fact, just one person on the team could do that. Then, once the first binding is removed using the steps below, the local branches can be published (i.e., pushed) to the remote Central repository.

3. Each author on the team should then complete the following steps:
    a. Select **Project > Project Properties**.

       You can also remove the first binding by using the Settings view of the Source Control Explorer.
    b. Select the **Source Control** tab.
    c. Next to the **Remotes** field, click ▪▪▪.
    d. Select the **origin** row (which represents your first binding) and click **Remove**. This leaves just the Central binding.
    e. Click **OK**.
    f. In the Project Properties dialog, click **OK**.

> NOTE Your Git commit history will be retained when you move from a dual-bound configuration to a single-bound setup. That's because the history is recorded in your local Git repository where your Flare project exists. Removing the first binding will have no effect on the commit history.

How to Move From Dual-Bound to Single-Bound for Non-Git Providers

If your first binding is to a non-Git provider (Perforce Helix Core, Subversion, Team Foundation Server), complete the following steps in your Flare project.

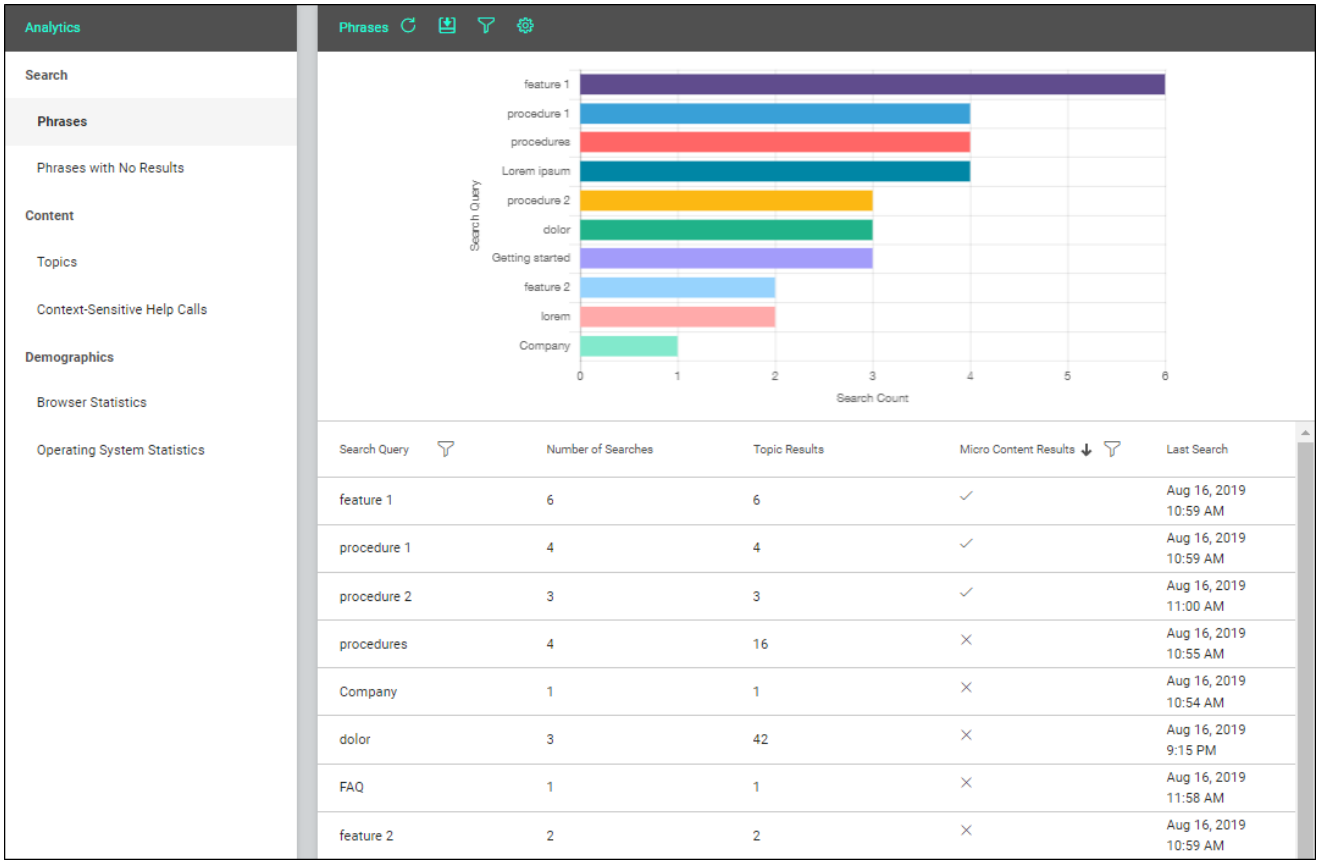
1. Each author on the team should commit all outstanding changes and synchronize.

   > NOTE Since Central does not support branches or streams from non-Git providers, keep in mind that you will not be able to move those up to Central. However, you could begin creating new branches in Flare and pushing those up to Central.

2. Each author on the team should then complete the following steps:
   a. Select **Project > Project Properties**.
   b. Select the **Source Control** tab.
   c. In the section showing your first binding, click **Unbind Provider**. (The second binding to Central is shown in a separate section; leave that binding alone.)
   d. Click **OK**.
   e. In the Project Properties dialog, click **OK**.

Central Analytics

If you have a MadCap Central license, you can view analytics on published Flare HTML5 output. This includes search phrases used, search phrases with no results, topics viewed, context-sensitive Help calls, and demographic statistics (browsers and operating systems).



Search Query	Number of Searches	Topic Results	Micro Content Results	Last Search
feature 1	6	6	✓	Aug 16, 2019 10:59 AM
procedure 1	4	4	✓	Aug 16, 2019 10:59 AM
procedure 2	3	3	✓	Aug 16, 2019 11:00 AM
procedures	4	16	×	Aug 16, 2019 10:55 AM
Company	1	1	×	Aug 16, 2019 10:54 AM
dolor	3	42	×	Aug 16, 2019 9:15 PM
FAQ	1	1	×	Aug 16, 2019 11:58 AM
feature 2	2	2	×	Aug 16, 2019 10:59 AM

General Information

- "Types of Analytics Information" on the next page
- "Host Output Anywhere" on page 43

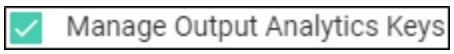
Main Activities

- "Creating and Editing Analytics Keys" on page 113
- "Associating Analytics Keys With Targets" on page 114
- "Managing Analytics Keys" on page 115
- "Downloading Analytics" on page 116

> NOTE If the project *is not bound to Central*, you can view the analytics information on any Central license where you are using the analytics key. However, if the project *is bound to Central*, you can view its analytics only on the Central license where the project is uploaded.

Permission Required?

To create and manage keys, you must have the following permission setting in Central:



However, all users on the license can view analytics data, regardless of their permission settings.

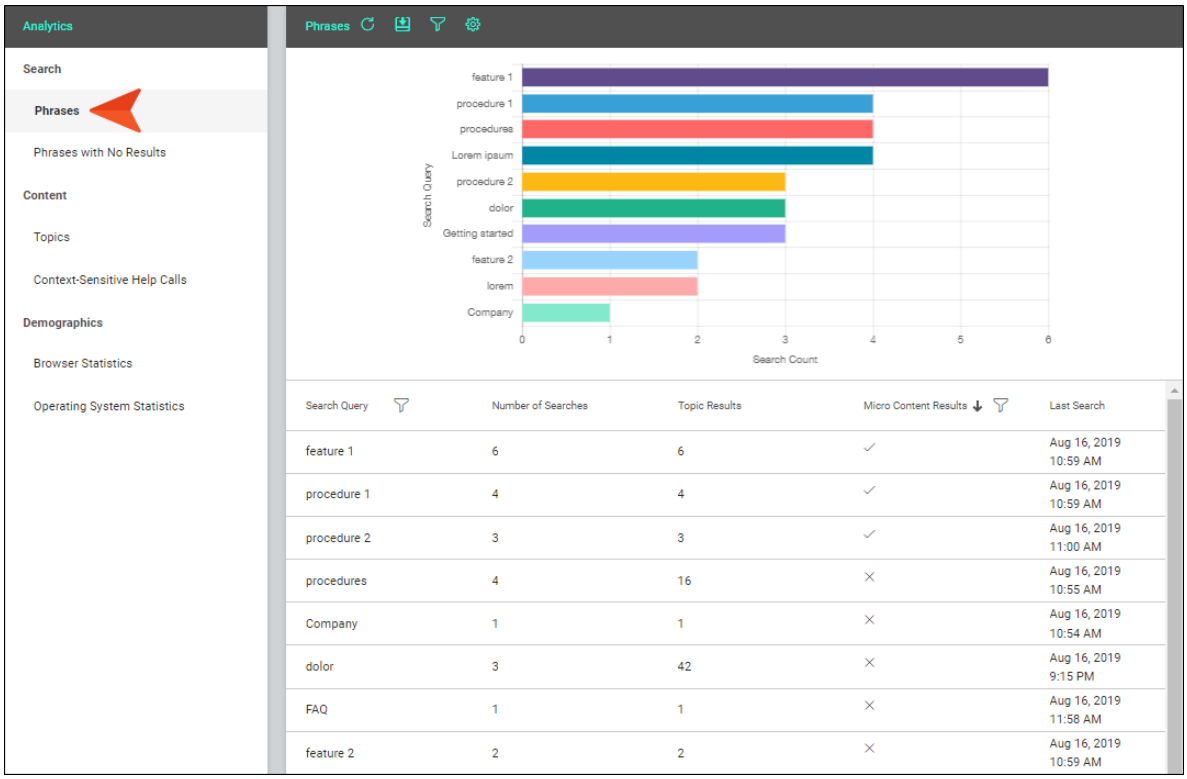
For more information about permissions, see the Central online Help.

Types of Analytics Information

By using Central Analytics, you can obtain various types of information about your output.

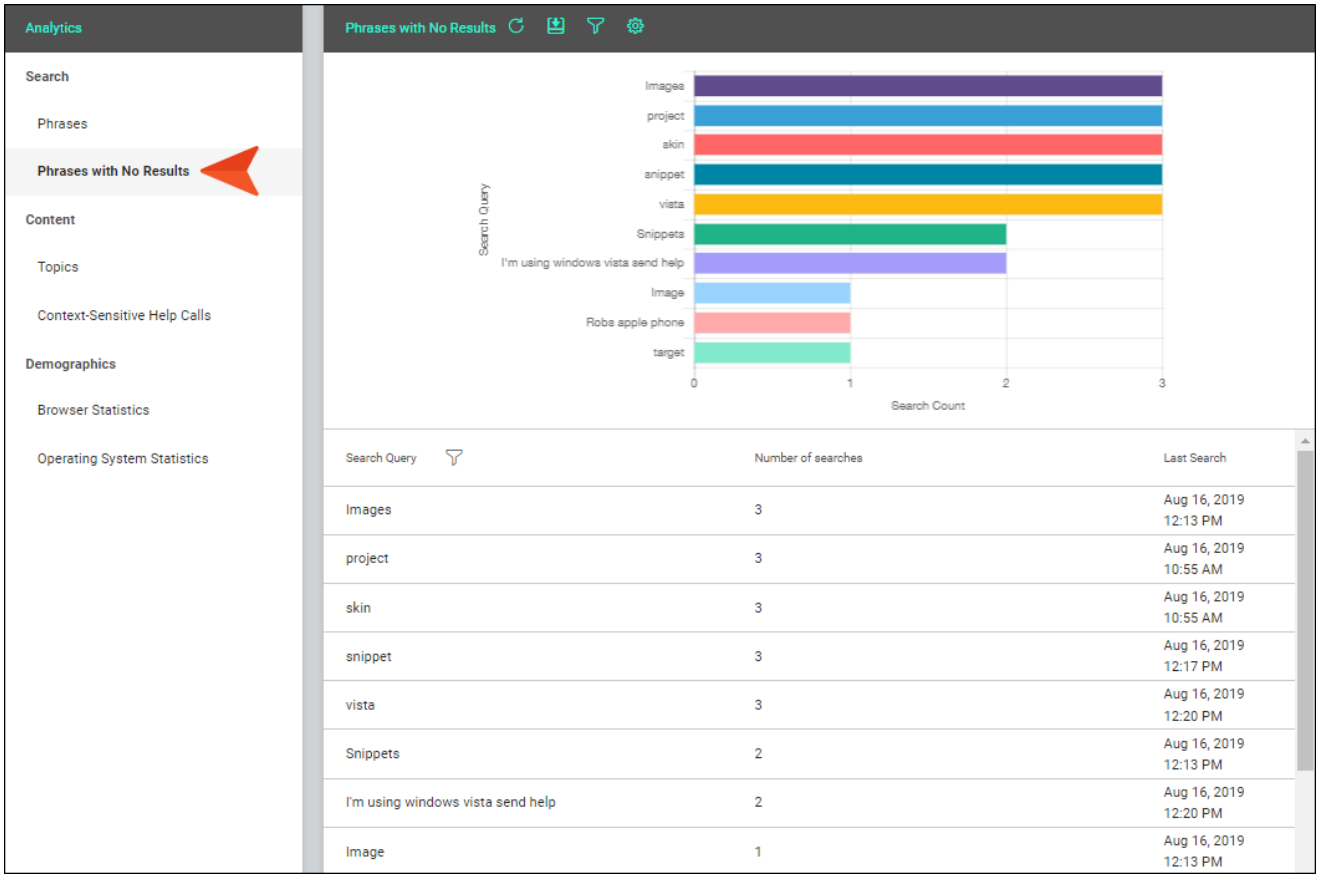
Phrases

Shows all search queries made to the output, including whether a search term returned any micro content results. *This is useful because it indicates the kind of documentation most often sought by users, as well as whether you should create new micro content to match certain search terms.*



Phrases With No Results

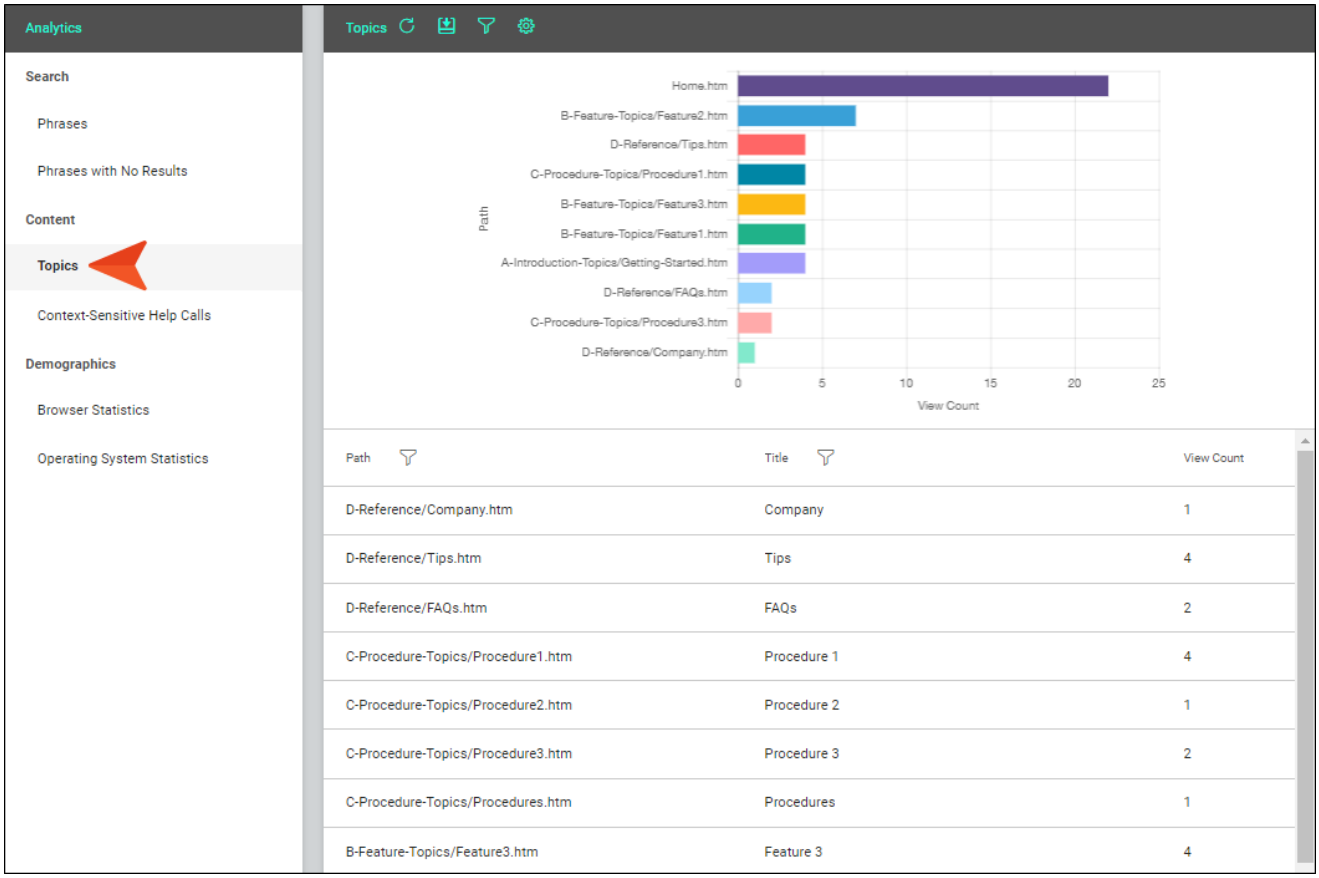
Shows all search phrases that didn't produce any results. *This is useful because it indicates if you need to produce new documentation to provide results for users, or perhaps create synonyms that will show results for existing documentation.*



Search Query	Number of searches	Last Search
Images	3	Aug 16, 2019 12:13 PM
project	3	Aug 16, 2019 10:55 AM
skin	3	Aug 16, 2019 10:55 AM
snippet	3	Aug 16, 2019 12:17 PM
vista	3	Aug 16, 2019 12:20 PM
Snippets	2	Aug 16, 2019 12:13 PM
I'm using windows vista send help	2	Aug 16, 2019 12:20 PM
Image	1	Aug 16, 2019 12:13 PM

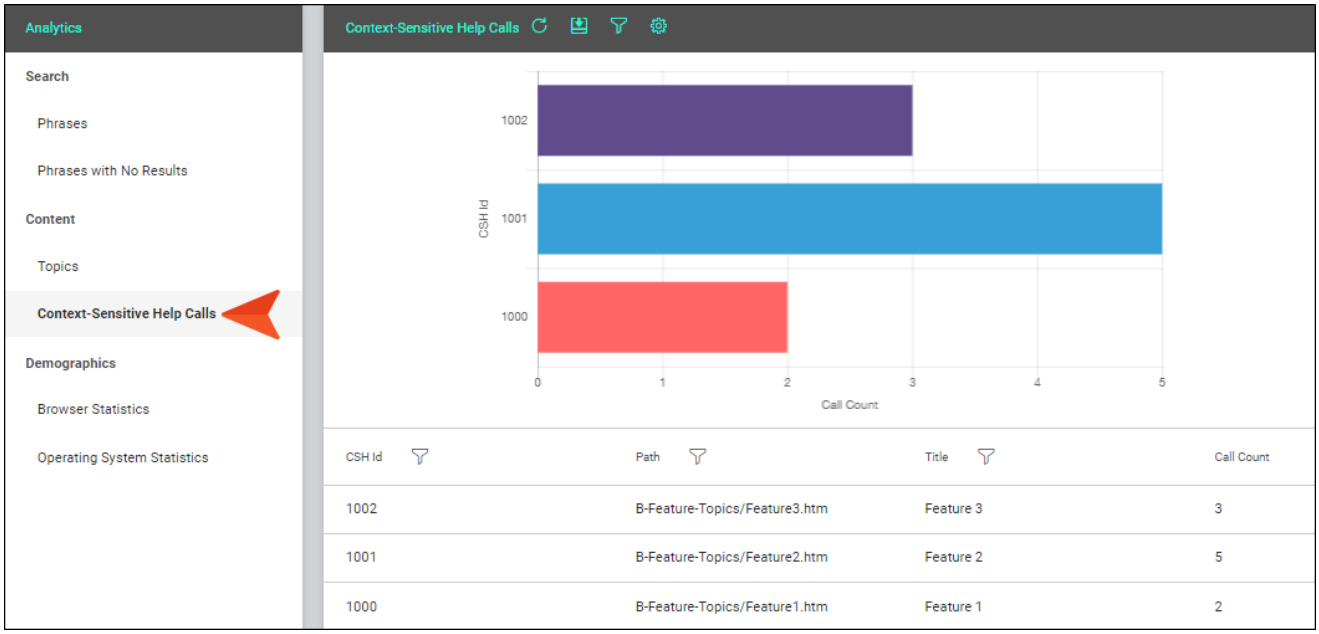
Topics

Shows all topics in the output that have been viewed at least once. *This is useful because it indicates the topics where you might want to focus much of your attention.*



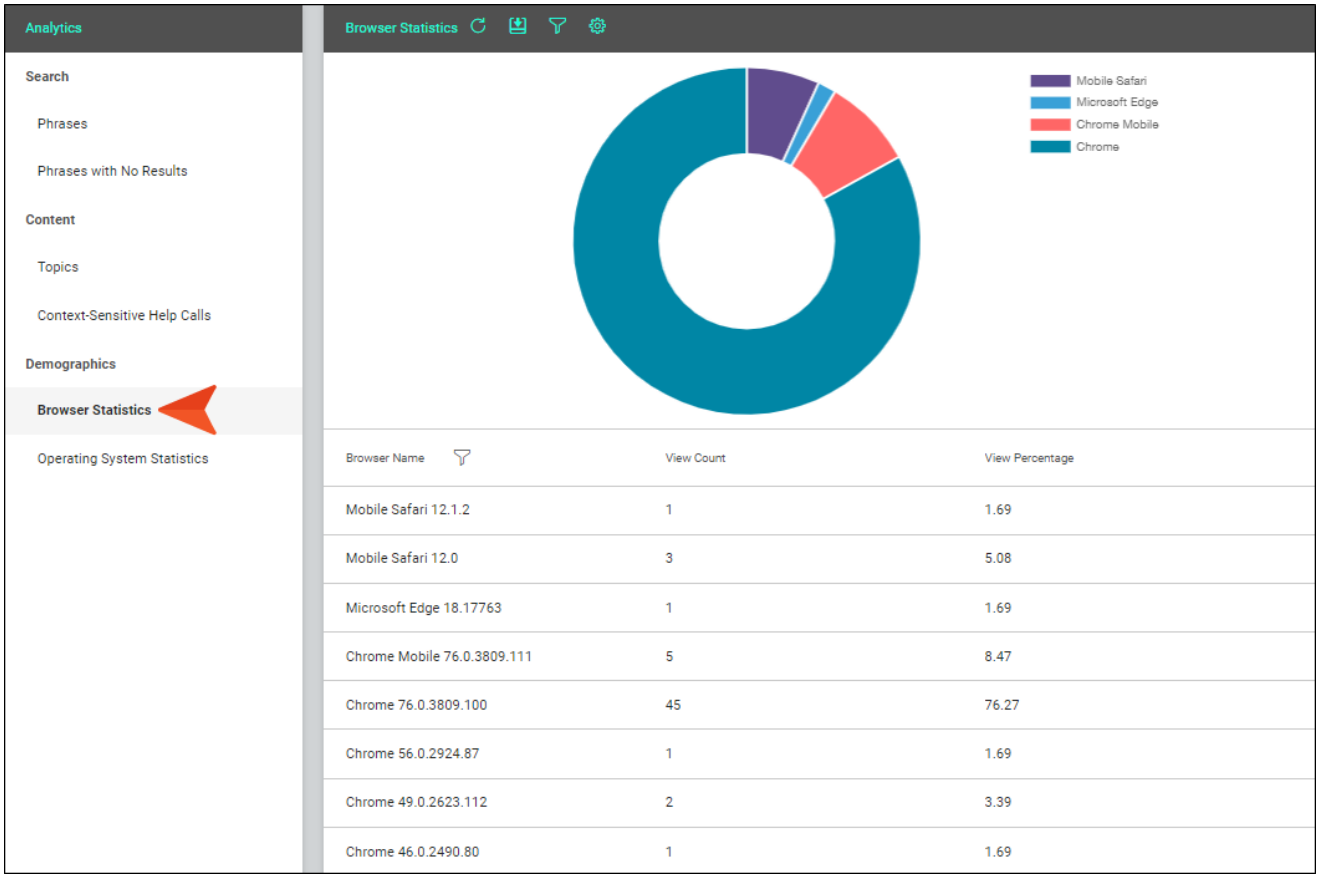
Context-Sensitive Help Calls

Shows context-sensitive Help identifiers that have been invoked at least once in the output. *This is useful because it indicates which areas of your product's interface are most active, and perhaps even unclear to users.*



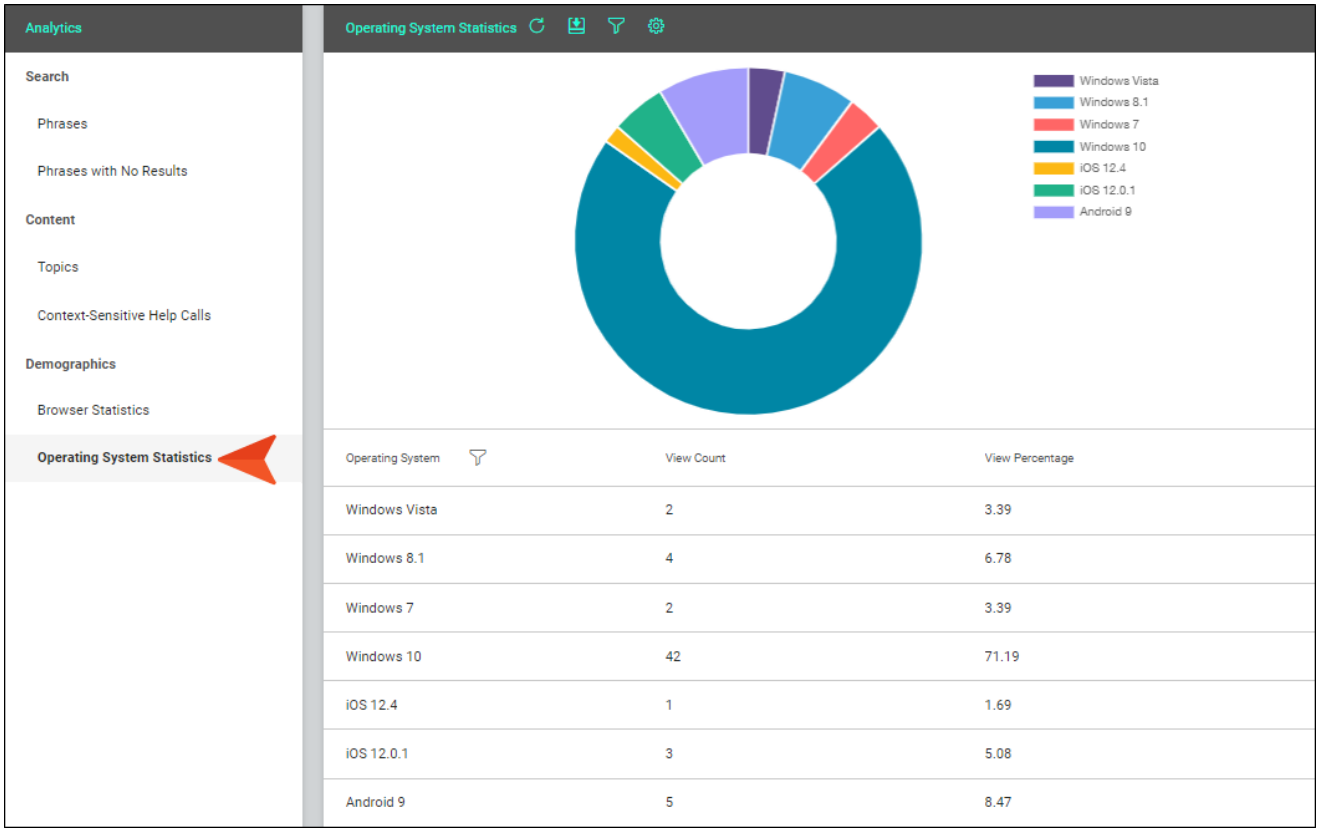
Browser Statistics

Shows which browsers are being used to view your online output. The graph shows a broad view of the browsers, and the grid breaks down the information into specific browser versions. *This is useful because it indicates which browsers you should focus on supporting.*



Operating System Statistics

Shows which operating systems are being used to view your online output. *This is useful because it indicates which operating systems you should focus on supporting. It might even be used to determine whether your company's software needs to continue supporting certain operating systems.*



Operating System	View Count	View Percentage
Windows Vista	2	3.39
Windows 8.1	4	6.78
Windows 7	2	3.39
Windows 10	42	71.19
iOS 12.4	1	1.69
iOS 12.0.1	3	5.08
Android 9	5	8.47

NOTE The graph at the top is limited to the first 10 items. The grid below shows all items.

NOTE For more information about using the Analytics page—including downloading and filtering data—see the Central online Help.

Host Output Anywhere

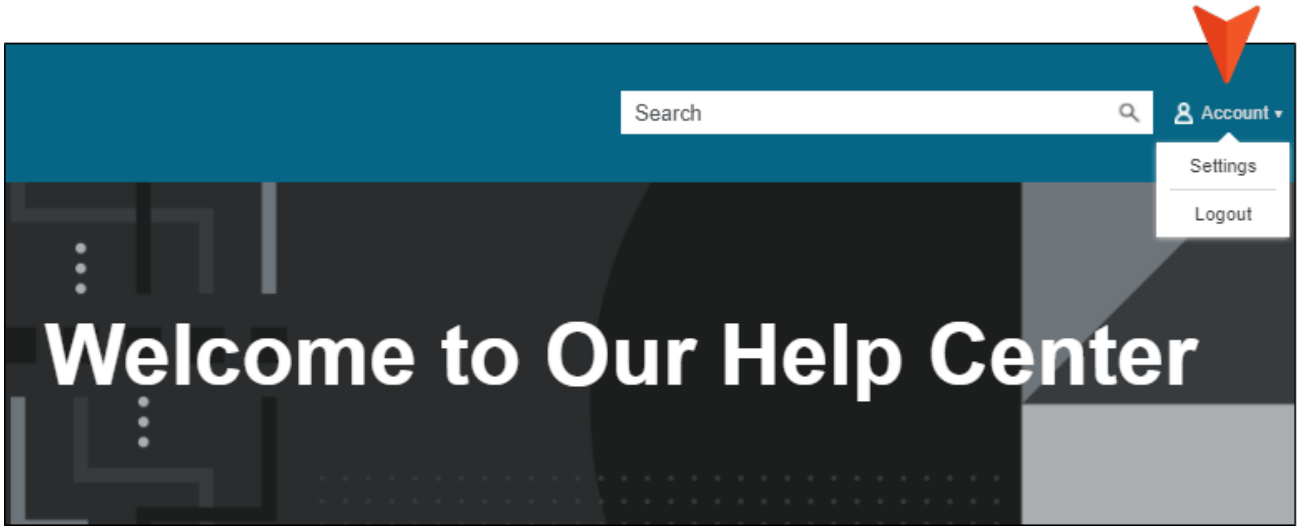
Central analytics works on Flare projects uploaded to Central, or you can host output on your own servers. If you host the output outside of Central, you still need to use Central with a key to view the analytics data. Also, the server where the output is hosted must be able to communicate with Central (e.g., not be behind a firewall).

Private Outputs on MadCap Central

Supported In:

You can produce private output that requires a user to log in with an email and Central password. Without credentials, people cannot see the output. It is possible to set private output for all output types supported by Central. Also, accessing private output is not limited to people in your company. Central has a free viewer user type that you can use to add any of your customers to your Central license, and then give them access to live private output.

After users log in to see private output, an account link displays in the output interface for HTML5 targets (not for other output types). If the output is not set to private, this link is not shown. When users hover over the link, a menu shows two options.



For more details about making output private, see the Central online Help.

How to Customize the Location of the Account Link

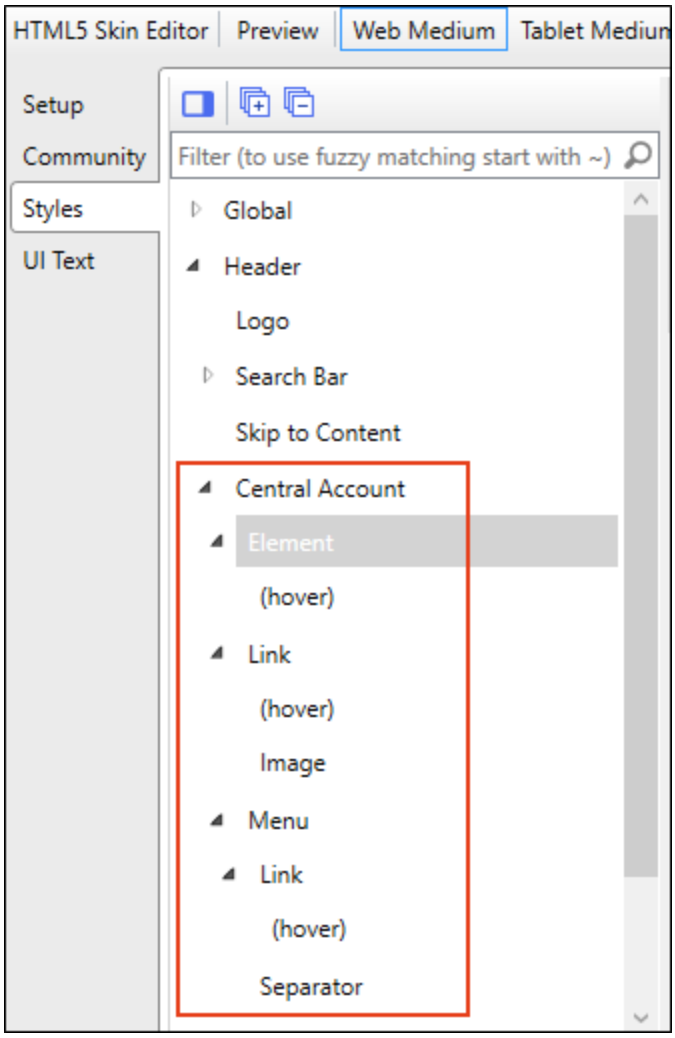
By default, the account link is included automatically in the header of the regular skin. However, if you want the account link to display somewhere else in the output, you can use a Central Account proxy and hide the link from the regular skin.

> NOTE You cannot use a proxy and skin component to customize the account link location for Tripane output. It is supported only for Side Navigation, Top Navigation, and skinless output.

1. (Optional) Add a Central Account skin component if you want to control the look of the account link in the output.
2. Open a topic (if you want the link to display on only one page) or template page (if you want the link to display on many pages).
3. In the topic or template page, place your cursor where you want the account link to be added. In a template page you may need to press the down arrow on your keyboard and then press ENTER at the correct location. Then select **Insert > Proxy > Insert Central Account Proxy**.
4. (Optional) In the dialog, you can select a skin component to affect the look of the account link.
5. Click **OK**.
6. Click to save your work.
7. If you inserted the proxy into a template page, make sure to associate the template page with a target.
8. Open the regular HTML5 skin, and set the account link to be hidden from the top of the output.
    a. In the local toolbar, make sure **Web Medium** is selected.
    b. On the left, select the **Styles** tab.
    c. In the style group section, expand **Header > Central Account**, and select **Element**.
    d. In the properties section, expand **Layout**, and set the **Display** field to **none**.
    e. Click to save your work.

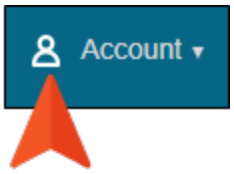
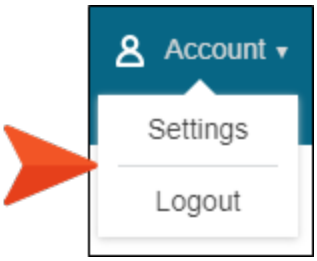
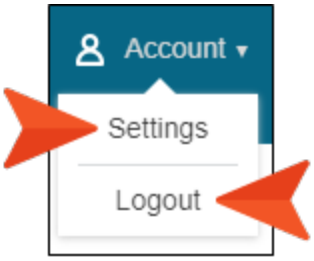
How to Change the Look of the Account Link

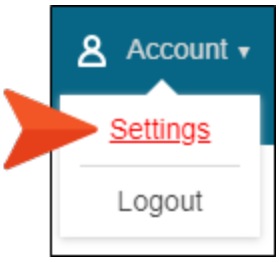
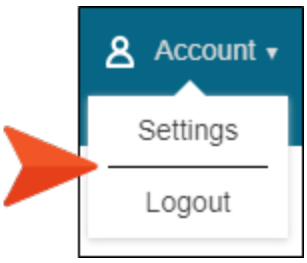
1. Open a regular HTML5 skin, or a Central Account skin component if you are using a proxy.
2. In the local toolbar, select the appropriate medium—**Web**, **Tablet**, or **Mobile**.
3. Select the **Styles** tab.
4. If you opened a regular skin, in the style group section, expand **Header > Central Account**.



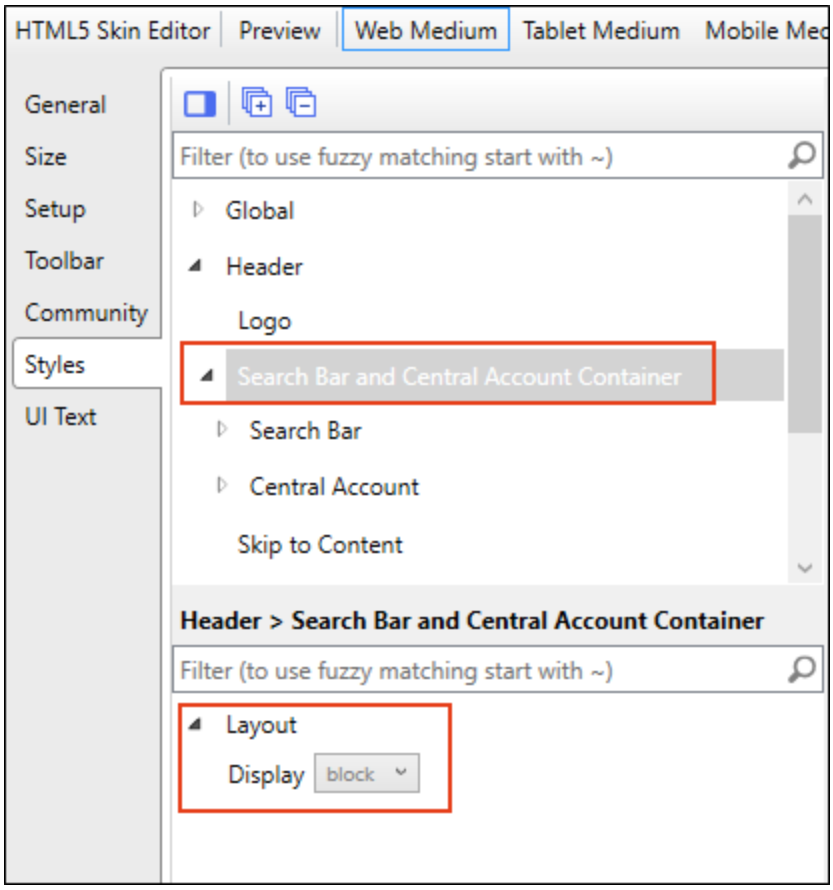
Then click any of the styles.

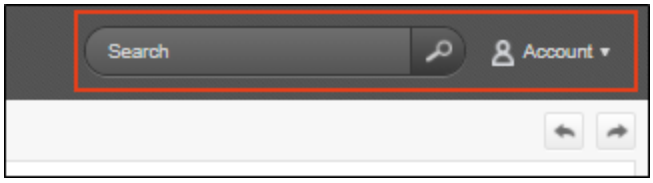
Header > Central Account > Element	This controls the look of the entire account link area.
Header > Central Account > Element > (hover)	This controls the look of the entire account link area when a user hovers over it.
Header > Central Account > Link	This controls the look of the area showing the image, text, and down arrow that the user first sees.
Header > Central Account > Link > (hover)	This controls the look of the area showing the image, text, and down arrow when a user hovers over it.

Header > Central Account > Link > Image	This controls the image shown to the left of the word "Account." 
Header > Central Account > Menu	This controls the look of the entire menu area. If you click this area in the preview to display the drop-down options, the popup will continue to display until you click again. This means you can continue to view the options while you make adjustments to the styles, allowing you to see your changes immediately. 
Header > Central Account > Menu > Link	This controls the look of the link for individual menu items. 

Header > Central Account > Menu > Link > (hover)	This controls the look of the link for individual menu items when a user hovers over them. 
Header > Central Account > Menu > Separator	This controls the look of area that separates menu items. 

For Tripane skins there is an additional style called "Search Bar and Central Account Container." This style lets you hide or show the container holding the search bar and Central account link.





5. Expand any of the nodes and complete the necessary fields. The properties and fields that you see are different for each node. Also, you might see nodes in the full skin that you do not see in a skin component, and vice versa. Following is a general list in alphabetical order:

ABSOLUTE POSITION

For certain styles, this can allow you to be more precise with the position of an element. This might be necessary to prevent the element from disappearing from the screen. In addition to entering values with specific units of measurement (e.g., pixels), you can enter auto in a field. You might do this, for example, if you want the left or right property to have a particular value, but leave the other property to be adjusted automatically.

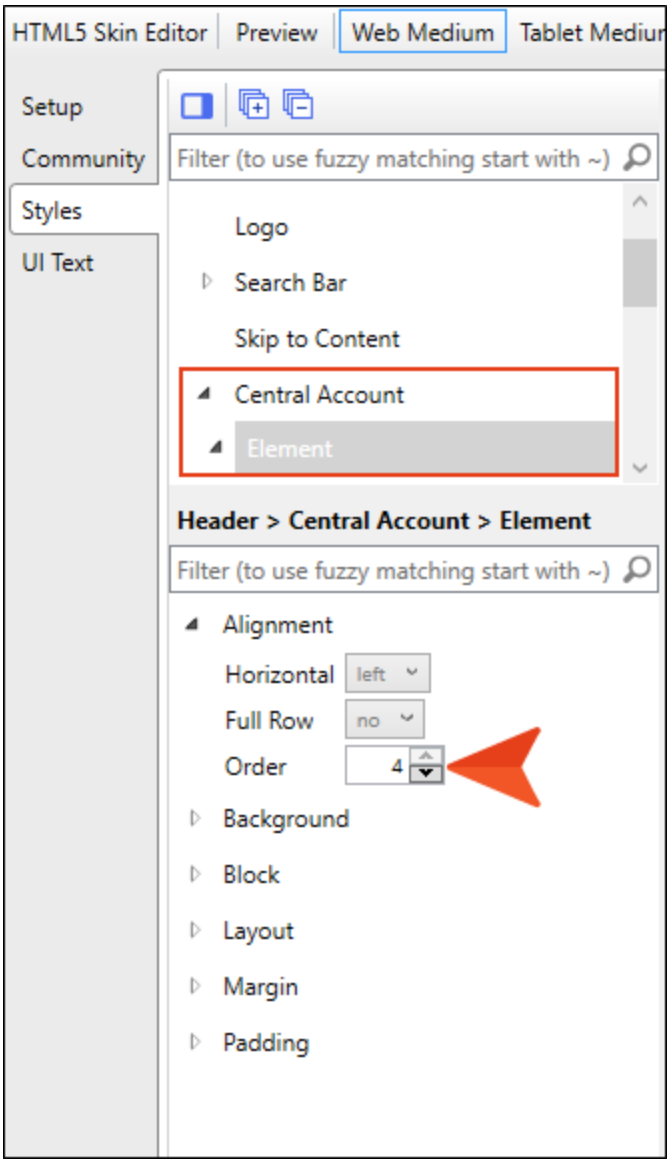
ALIGNMENT

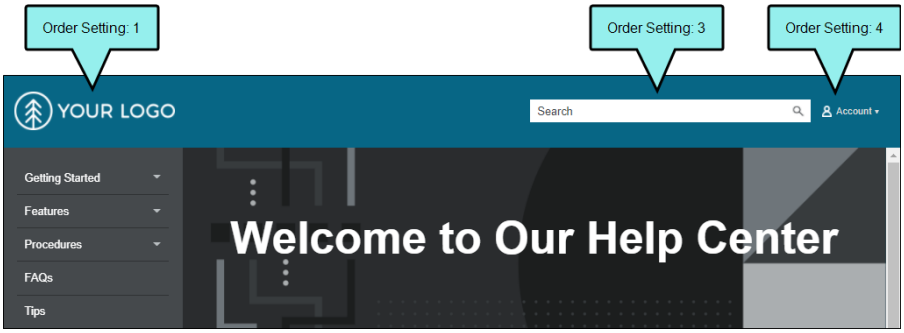
Depending on the style and element you are working with, you may be able to select one or more of the following:

- Horizontal This option lets you choose whether to position the element to the left, middle, or right.
- Vertical This option lets you choose whether to position the element on the top, middle, or bottom.
- Full Row This option allows the element to take up all of the space horizontally in the header.
- Order This option determines which element comes first (1), second (2), third (3), or fourth (4) in the header. Side Navigation has three elements (logo, search bar, and Central account link), but Top Navigation has four (logo, search bar, menu, and Central account link).

EXAMPLE

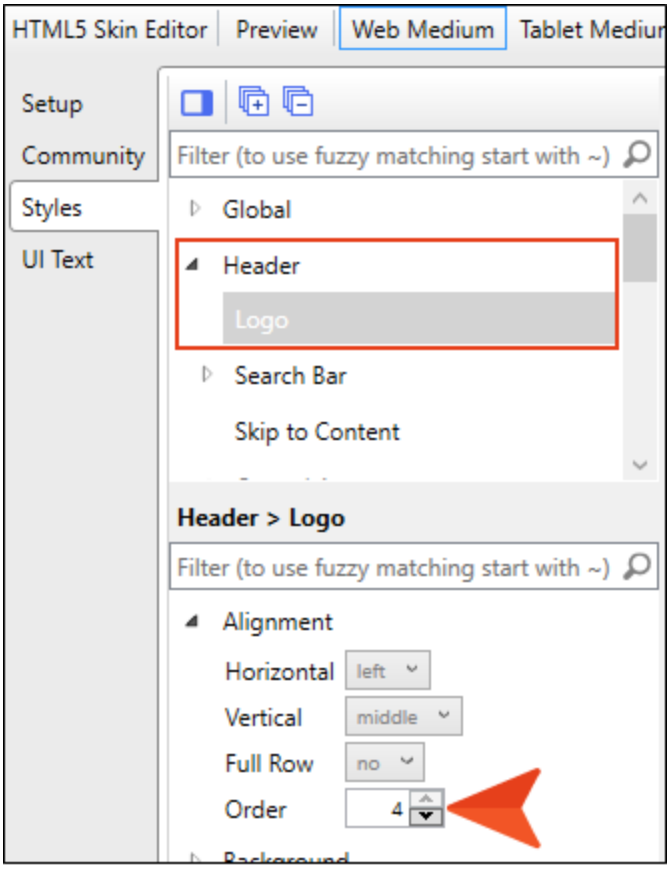
If you have a Side Navigation skin, the Central Account Element is set to have an order value of 4. This means it appears to the right of both the logo and the search bar, which have lower values for that option (the logo is set at 1, and the search bar is set at 3).

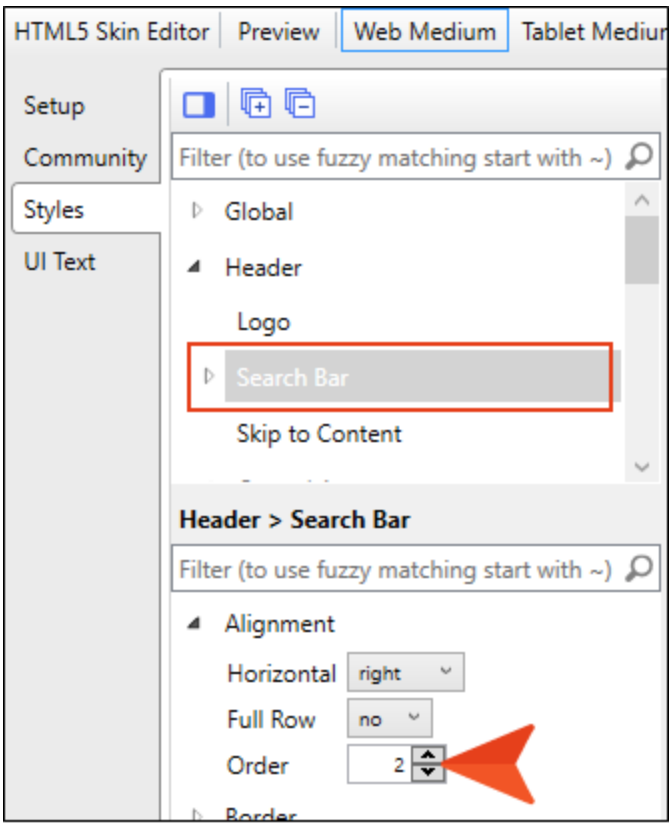


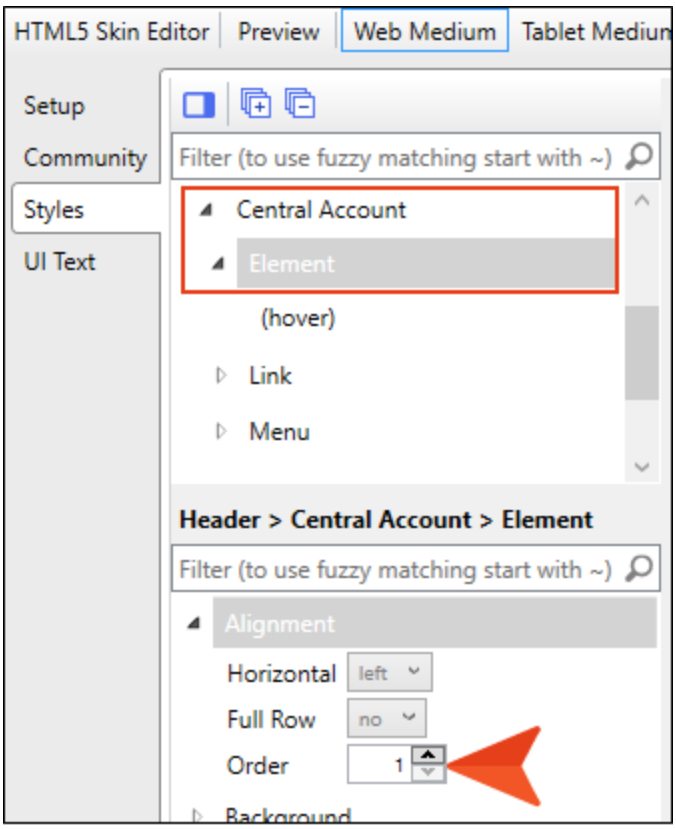


You change the values for the Alignment > Order property as follows:

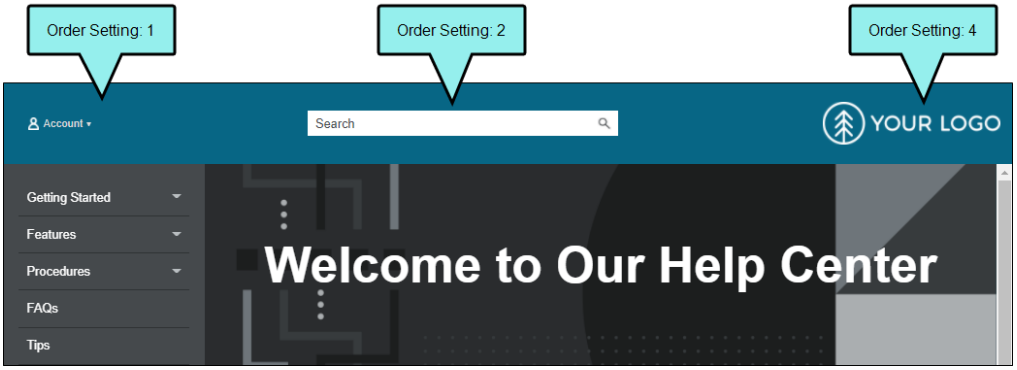
- Logo: 4
- Search Bar: 2
- Central Account Element: 1


HTML5 Skin Editor
Preview
Web Medium
Tablet Mediur
Setup
Community
Styles
UI Text
Filter (to use fuzzy matching start with ~)
Global
Header
Logo
Search Bar
Skip to Content
Header > Logo
Filter (to use fuzzy matching start with ~)
Alignment
Horizontal
left
Vertical
middle
Full Row
no
Order
4


HTML5 Skin Editor
Preview
Web Medium
Tablet Mediur
Setup
Community
Styles
UI Text
Filter (to use fuzzy matching start with ~)
Global
Header
Logo
Search Bar
Skip to Content
Header > Search Bar
Filter (to use fuzzy matching start with ~)
Alignment
Horizontal
right
Full Row
no
Order
2


HTML5 Skin Editor
Preview
Web Medium
Tablet Mediun
Setup
Community
Styles
UI Text
Filter (to use fuzzy matching start with ~)
Central Account
Element
(hover)
Link
Menu
Header > Central Account > Element
Filter (to use fuzzy matching start with ~)
Alignment
Horizontal
left
Full Row
no
Order
1
Background

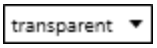
As a result, the header will look like this in the output:



BACKGROUND

You can change any of the following to affect the background:

- **Gradient** This lets you change the background color using a progression effect. Select the beginning gradient color in the first field, and then select a second gradient color in the second field. You can use any of the following to enter or select a color.

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways. Note that you can select a CSS variable.
	Adds a bar above the cursor. As you move the cursor over any area of your screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is loaded into the Color Picker dialog.

- **Color** This lets you select a single color.
- **Image** This lets you select an image for the background. Click and use the dialog to select an image file.
- **Repeat** Use this field to tell Flare whether the image should repeat or not.

> **NOTE** The Image field has the highest precedence, which means that if you enter settings in all of the Background fields, the image will win. Gradient has the next highest precedence. If you want to use the Color field, you need to make sure the Gradient fields are set to transparent and that there is no image selected.

BLOCK

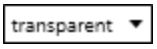
This is the container (or "block") holding content in an element. You can change any of the following to affect the block:

- Line Height This is how tall the container is that holds the content. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- Text Align This changes the alignment of the content. Click in the field and select an option (center, justify, left, right).
- Text Indent This is how far the text is moved inward. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- Word Wrap You can click in the Word Wrap field and choose whether to wrap text for the element (normal) or not (nowrap).

BOX SHADOW

You can change any of the following to affect the box shadow on the element:

- Horizontal Shadow Length This is the position of the horizontal shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- Vertical Shadow Length This is the position of the vertical shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- Shadow Blur Length This is the blur distance. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- Shadow Color This lets you select a color for the shadow. You can use any of the following:

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways. Note that you can select a CSS variable.
	Adds a bar above the cursor. As you move the cursor over any area of your screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is loaded into the Color Picker dialog.

- Shadow Inset This changes the shadow from an outer shadow (outset) to an inner shadow. Click in the field and select an option. If you select **false** the shadow will be outset. If you select **true** the shadow will be inset.

FONT

You can change any of the following to affect the font:

- Color This lets you select a color for the font. You can use any of the following:

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways. Note that you can select a CSS variable.
	Adds a bar above the cursor. As you move the cursor over any area of your screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is loaded into the Color Picker dialog.

- Family You can select or type a specific font family (e.g., Arial, Tahoma, Verdana).
- Size You can change the font size. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points).
- Weight You can click in this field and make the font normal or bold.
- Style You can click in this field and make the font normal or italic.

HEIGHT

You enter a height for the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

LAYOUT

You can click in the **Display** field and choose one of the options (some of these options might not be available for certain styles):

- **none** The element will not be shown.
- **block** The element will be shown and adds a line break after it. For some elements this option might be called **flex**, which reflects the flexbox technology associated with it.
- **inline** The element will be shown but does not add a line break after it, allowing it to be positioned next to other elements.
- **inline-block** The element will be shown but does not add a line break after it, allowing it to be positioned next to other elements. This is similar to inline; however, with this option, a width and height can be set on the element, and top/bottom margin and padding settings are respected.

MARGIN

You can change the margin for any of the sides on the element (Left, Right, Top, Bottom). In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

PADDING

You can change the padding for any of the sides on the element (Left, Right, Top, Bottom). In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

SIZE

You can change the height or width of the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

TEXT DECORATION

You can click in the **Decoration** field and select a text decoration (e.g., underline, line-through, overline, underline).

6. Click to save your work.

Flare Features Not Supported in MadCap Central

There are certain features in Flare that are not fully supported when you generate targets from MadCap Central. However, a few of these features are supported when publishing directly to Central from a local Flare project.

Automatically Get Latest Files

If you are working in a dual-bound scenario (see "Dual-Bound Projects" on page 16) and you select the option in the target to automatically get the latest version of files before generating, this option will not work when you build from MadCap Central. Instead, you must first synchronize your files with source control in Flare and then push those changes to Central.

Batch Targets

Batch targets cannot be generated from MadCap Central.

Custom Build Log Folder

In the Options dialog (File > Options), Flare has a feature that lets you choose a custom folder to hold the log file after generating output. This feature is not supported when building from MadCap Central. Instead, the log file is stored at the root level of the output folder.

Custom Fonts

If you are using custom fonts in your Flare project, they are not supported when you build output from MadCap Central.

Custom Output Folder

Flare targets let you choose a custom location for your output folder. However, this is not supported when building from MadCap Central. You must use the default location.

Date/Time Variables

Target generation occurs on a server with Central, so date/time variables are not supported.

Elasticsearch

Elasticsearch is one of the search engine options that you have in Flare (along with MadCap Search and Google Search). Currently, using Elasticsearch in your HTML5 output is not supported in targets generated from Central.

However, Elasticsearch is supported when publishing directly to Central from a local Flare project.

Global Ignored Warnings

You can tell Flare to ignore certain warnings so that they do not display as such in the Builds window pane when generating output. This can be done globally for all targets in all projects. You can also override those global settings for a specific target in a project. However, if you generate output from Central, global settings to ignore warnings are not supported; instead, you must ignore warnings in the target.

Linked Mimic Movies and Projects

When building from MadCap Central, external links to Mimic movies/projects will not work. However, embedded movies in a Flare project are supported.

MadCap Pulse

MadCap Pulse is not supported in output that is generated from Central. In order to use Pulse with your output, it must be hosted outside of Central.

Pre- and Post-Build Events

If you create pre- or post-build events in your Flare target, they will not function when building from MadCap Central.

However, pre- and post-build events are supported when publishing directly to Central from a local Flare project.

Publishing to Outside Servers

Flare lets you create destination files for the purpose of publishing to external servers. This feature is not supported when building from MadCap Central. Instead, live builds are always published in Central.

Viewing Some Output Formats

The following types of output can be generated from MadCap Central, but they cannot be viewed (output files are downloaded to your computer):

- DITA
- Eclipse Help
- EPUB
- Microsoft HTML Help

  > NOTE Due to issues with Microsoft HTML Help, the CHM file that is downloaded will initially be blocked. To unblock the CHM file, remove it from the ZIP file. Then right-click the CHM file and select **Properties**. On the **General** tab in the dialog, select **Unblock**.

- Microsoft Word
- Server-Based Outputs:
  - HTML5 Plus
  - WebHelp Plus

  > NOTE If you want to produce server-based output from MadCap Central, you will need to download the output from Central after it finishes building. You can then publish the output to your server.

Security and MadCap Central

MadCap Central leverages the security, power, and flexibility of the cloud to mitigate or eliminate many of the technical hurdles faced by both content creators and information technology professionals. The overhead traditionally associated with managing complex systems can hinder the ability to create content and deliver content efficiently. The goal of this is to provide a high-level overview of the ways that Central addresses these challenges.

Security and privacy are top priorities at MadCap Software, especially when providing customers with a cloud-based system in Central. We are committed to keeping your files, data, and communications secure. Therefore, in choosing Microsoft® Azure as a partner, we focused on selecting a security center that understands the importance of privacy and complies with the highest international and industry-specific compliance standards and uptime guarantees. Microsoft Azure regularly undergoes rigorous third-party audits to ensure and verify the highest level of security controls.

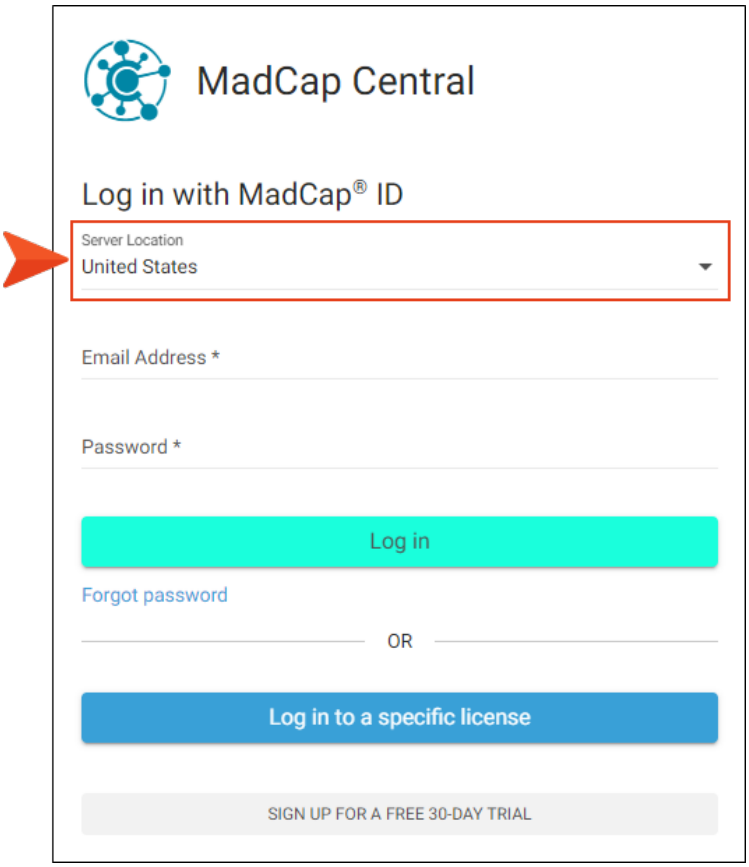
For more information, see the following:

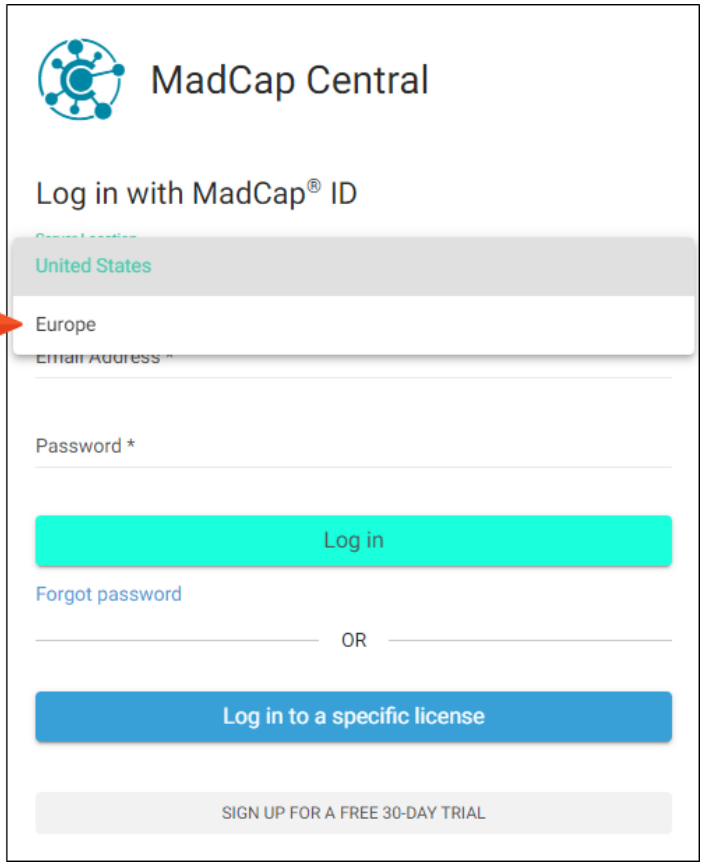
```
https://www.microsoft.com/en-us/TrustCenter/Security/default.aspx
```

Server Regions

New license subscriptions have the option a server region.

Depending on which region you chose when subscribing to a Central license (United States or Europe), you can select the corresponding server when logging in to Central.





- The European server is located in Germany.
- If you have an existing license (i.e., one that was created before the introduction of the European server), it will remain on the United States server.
- The server that you select will hold all of your data and hosted output for that license.
- Previous versions of Flare will not display the drop-down for selecting a server.
- Regardless of the server, the default MadCap domain is madcapcentral.com (e.g., fictionsoft.madcapcentral.com). Of course, you can still add your own host mapped domain (e.g., help.fictionsoft.com).
- Each server will have access to a global list of vanities, but will not contain any sensitive information. License vanity, region, and Central key are the only pieces of data shared globally, so each server has access to that data.

Data Centers and Disaster Recovery

MadCap Software partners with Microsoft Azure to securely host mission-critical infrastructure, such as MadCap Central. Depending on the server you choose when subscribing to Central, the data residency and infrastructure are hosted in one of two locations:

- United States Azure WESTUS datacenters
- Europe Azure German datacenter

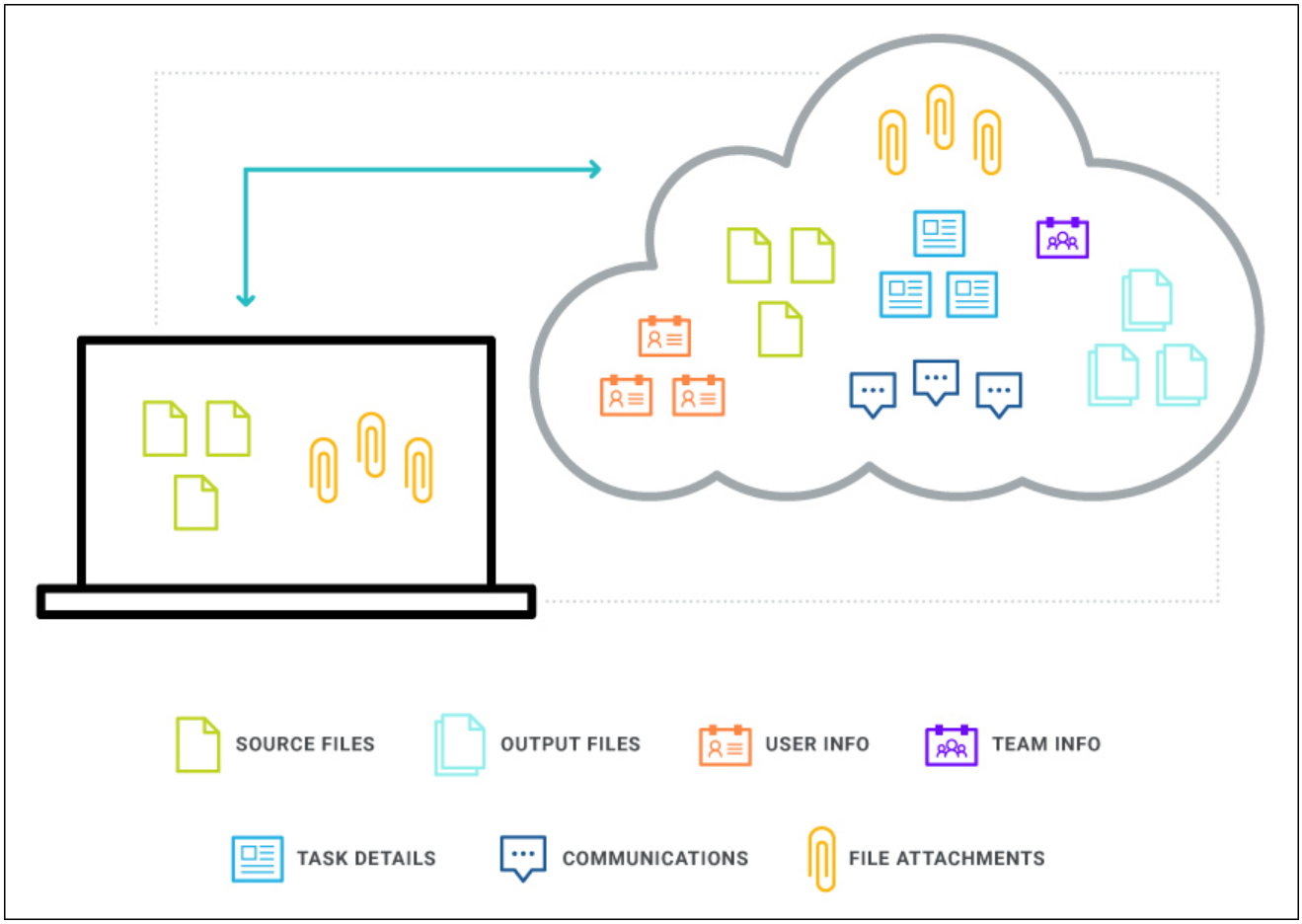
These Azure data centers are ISO27001 certified. For details, see:

```
https://www.microsoft.com/en-us/cloud-platform/global-datacenters
```

In addition to data stored in the database, the source control provider (Git) has the ability to failover in case of an outage. This is essential in handling mission critical data in the event of a catastrophe.

Source Control Provider and Data Storage

MadCap Flare projects can be uploaded to Central, thus creating copies of those projects on the cloud. From the cloud versions of your projects, output files can be generated and published. In addition to Flare source and output files, Central hosts many other types of data, including user and team information, task details, communications between individuals, and ancillary file attachments.



When Flare projects are uploaded to a Central license, they are bound by a Git source control connection.

The following objectives are important for this relationship:

- Easy Access Projects are hosted in the cloud, which means they can be accessed from anywhere.
- Security Projects are stored securely in the cloud. Communication between your local desktop and the cloud is over HTTPS (HTTP over SSL). This is an encrypted, secure channel of communication. Source control data is secured by user name and security token. It is also possible to configure a license so that authentication is based on single sign-on (SSO) from the company's identity provider, thus adding another level of security.
- Data Transference Users can synchronize the local and cloud versions of Flare projects after changes are made to files. In addition, you can retrieve files to a local machine whenever necessary. Uploaded projects can be imported and generated output files can be downloaded from Central.
- Storage MadCap Central stores user data in different ways depending on the nature of the data. Some examples of data storage include encrypted SQL databases, source control repositories (GIT), and Azure blob storage.
    - Sensitive data—such as user credentials and company information in the database tier—are stored using encryption to mitigate unauthorized access to data, and ensure your information is stored securely.
    - The methods of storing data follow industry security standards and offer a highly scalable solution to ever expanding storage management tracking needs.
    - If you choose to cancel your subscription, hosted data can be wiped at your discretion.
- Backups All data is backed up daily. All user data in Central database backups are encrypted, transmitted remotely, and stored securely. This helps prevent unauthorized access.

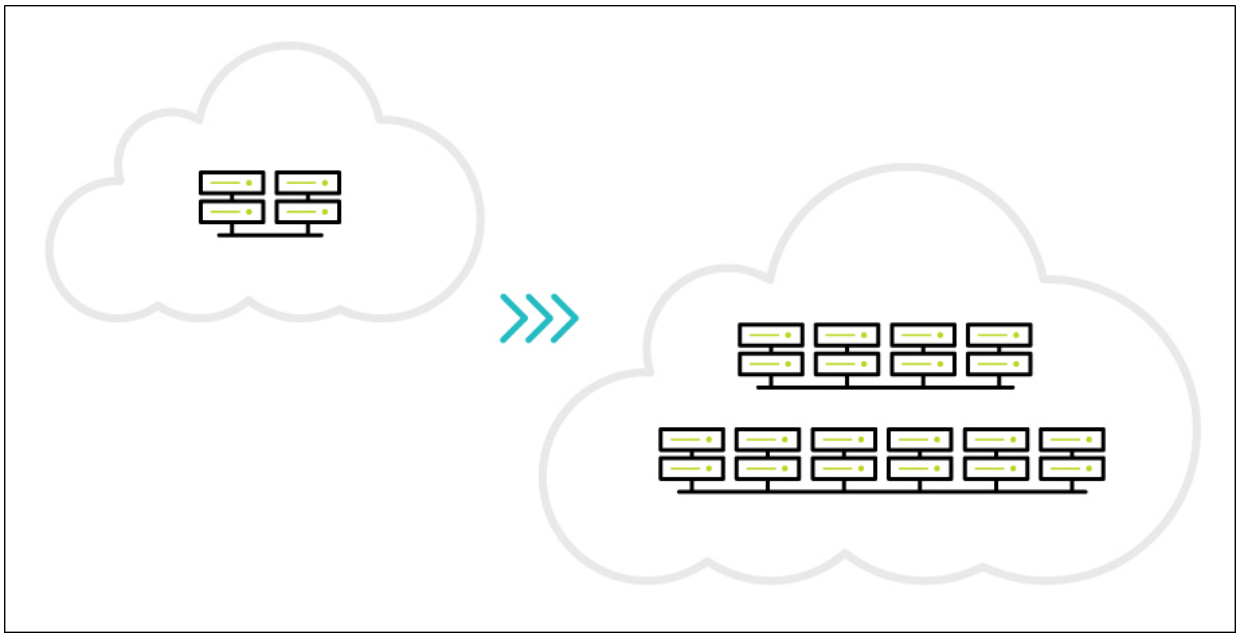
Web Server Hosting and Management

Every MadCap Central subscription includes access to our highly scalable cloud-based build server. Because MadCap Central is a cloud-based solution, content creators can rapidly generate and publish their content with minimal effort while benefiting from a high level of security, scalability, and durability, something that would otherwise require skilled IT resources to implement. Published output is hosted on a robust, secure, geographically redundant web application server.

With that in mind, the following are important considerations:

- Reduced Hosting Costs You can host all of your projects within Central and never worry about managing or scaling costly web servers. Whether you have a small, single project or several large projects, you can host and manage all of your content in one place.
- Automation and Convenience From Central, you generate and publish output with minimal effort. You also have the ability to quickly roll back published outputs when necessary, with the click of a button. Builds can be initiated manually or they can be scheduled. All of this can be done from any device supporting the browser-based interface.
- Availability and Monitoring Your content is vital to your company, so 24-hour availability of the web server is crucial, as is constant uptime monitoring. All components of MadCap Central are monitored 24 hours a day, 7 days a week, 365 days a year using remote monitoring systems. MadCap Software has staff on hand to respond to outages and security breaches.
- Security Management Data in MadCap Central is secured in the various ways.
    - Web services have built-in load balancers to stave off most DOS (Denial-Of-Service) attacks.
    - We provide unauthorized access monitoring and incident response. Any breaches in data security will be reported to affected customers.
    - User portal and API endpoints (i.e., the connection between Flare and Central) are secured via user authentication.
    - All data endpoints are secured using SSL. The user portal, API endpoints, source control provider (Git), and all backend system communications are all encrypted.
    - Published sites that are not set to “live” are secured by user authentication.

- **Scalability** As your content management needs grow, so will your demands for a system that dynamically scales as well. Over time, the builds that you generate on Central will consume more and more space. Central offers ease with scalability for your output, which helps to eliminate worries about CPU, memory, and disk-intensive projects.



Service Level Agreement

The full Service Level Agreement (SLA) for MadCap Central (and other MadCap Software products) can be found here:

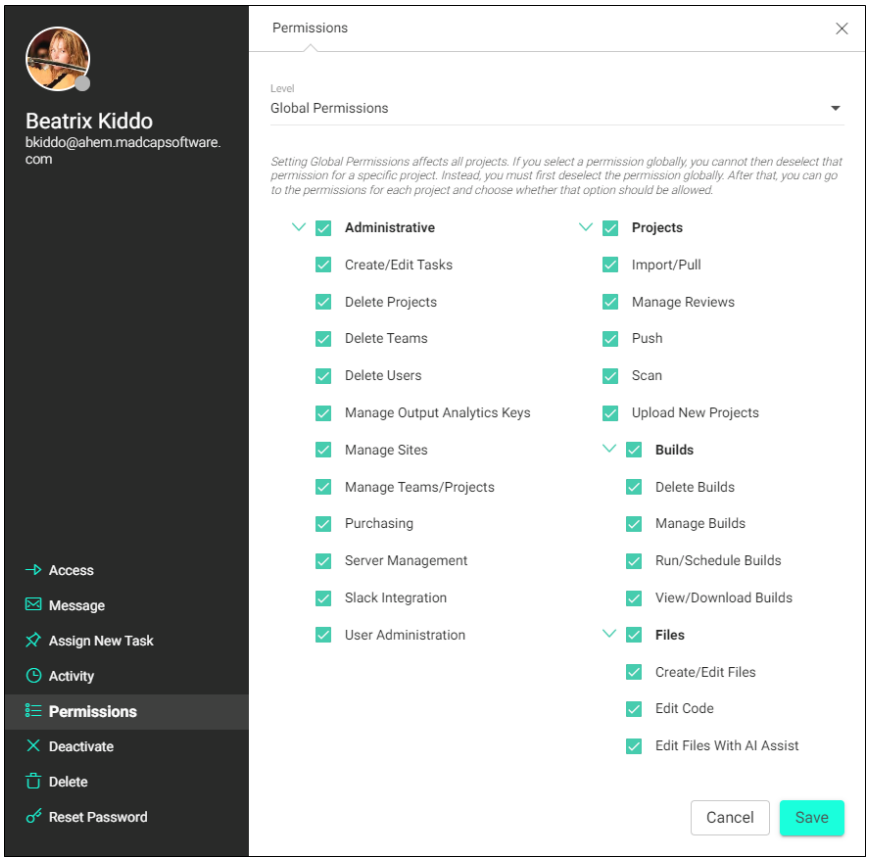
```
https://www.madcapsoftware.com/company/service-level-agreement/
```

Application and Browser

Much of the work in MadCap Central (e.g., management of users, teams, tasks, projects, and builds) requires user interactions to be performed through a browser. In addition to Flare requiring end-to-end encryption, all traffic from your web browser to the MadCap Central portal is also secured using industry standard security practices such as SSL.

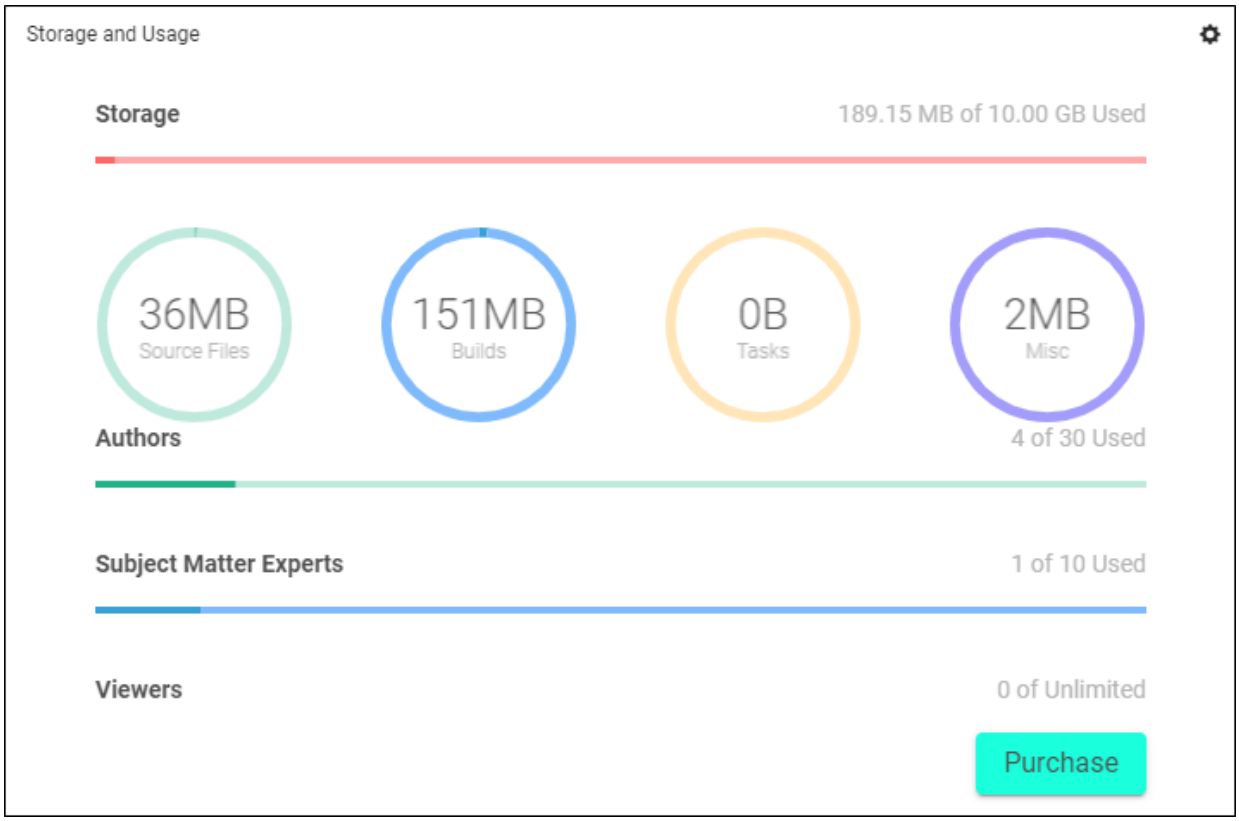
Following are some of the notable features in Central that contribute to a secure, accessible work experience:

- Permissions In Central, permissions allow you to dictate which users can see certain information and perform a variety of functions. In addition to basic permissions, there are several administrator-level rights that can be set, giving you the most flexibility to control how your data and files are being managed and distributed.

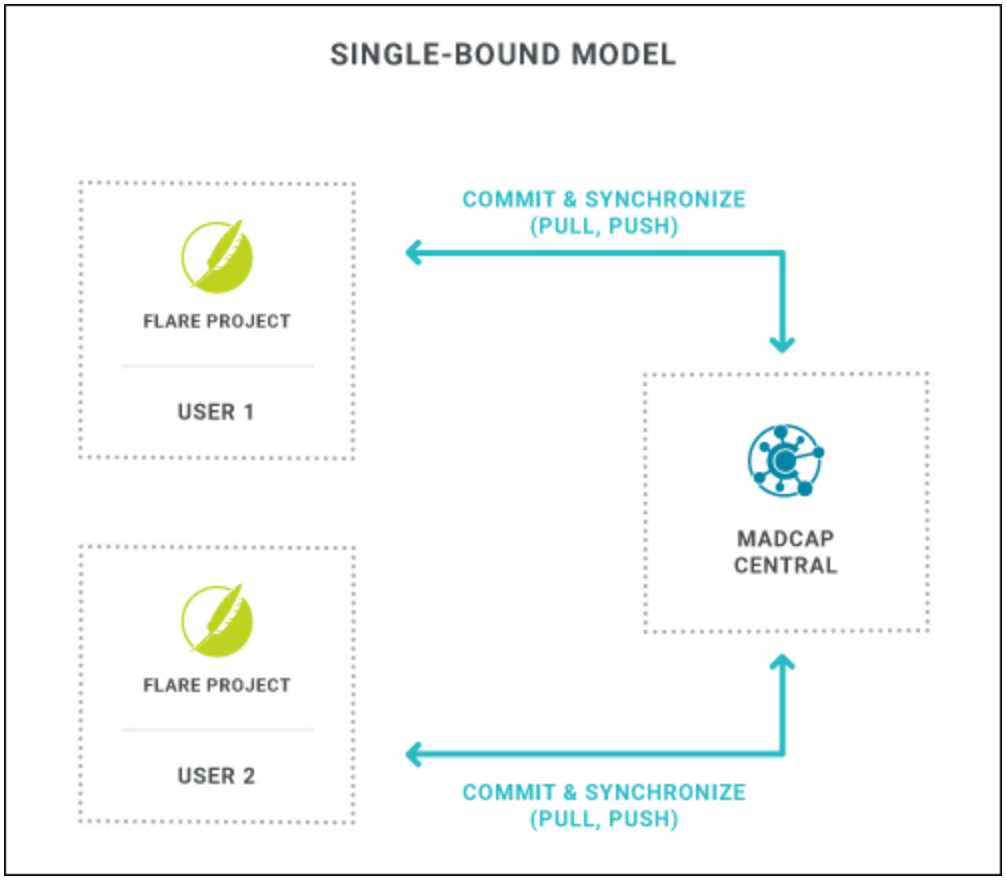


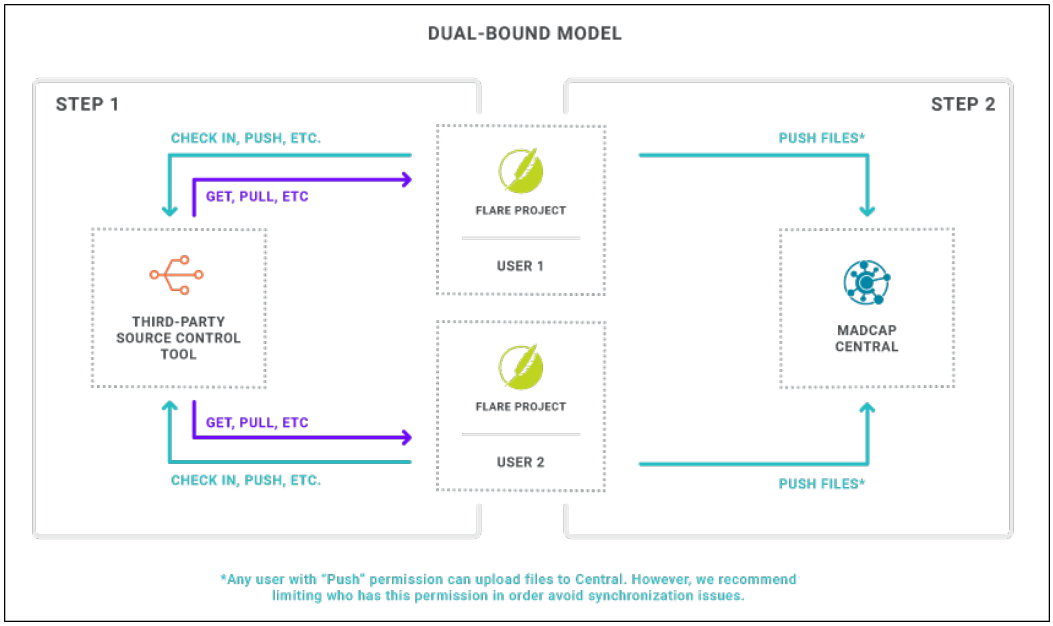
- Importing Projects If another user uploads a Flare project to your MadCap Central license and you do not yet have that project on your computer, you can import it. You can do this from the MadCap Central window pane in Flare.

- Downloading Builds After generating a target, you can download the output to your computer. This is an optional step, in case you want to have a local copy of the build (e.g., you want to view your Microsoft HTML Help output, which is an output format that you cannot view from Central).
- Purchasing Seats and Space Built-in widgets inform you when space or user count becomes a concern. When you run out of user seats or storage space in Central, you can purchase more. You do not need to contact MadCap Software to do this; instead, you can purchase user seats and storage space directly from the Central interface.

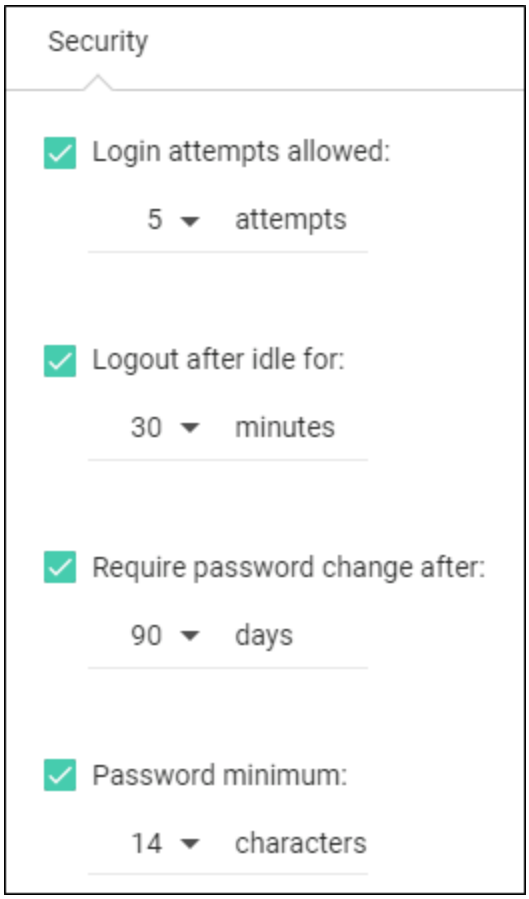


- Single-Bound and Dual-Bound Projects When you upload a Flare project to Central, the files are connected to Central via an integrated source control system (Git). Your interaction with source control can follow one of two models—single-bound (recommended) or dual-bound. Single-bound projects are not bound to an additional third-party source control provider; they only use Central's source control system. Dual-bound projects, on the other hand, are already bound to another source control provider, and therefore are bound to both the original third-party source control provider and to Central.




DUAL-BOUND MODEL
STEP 1
STEP 2
CHECK IN, PUSH, ETC.
GET, PULL, ETC
THIRD-PARTY SOURCE CONTROL TOOL
FLARE PROJECT
USER 1
FLARE PROJECT
USER 2
GET, PULL, ETC
CHECK IN, PUSH, ETC.
PUSH FILES*
MADCAP CENTRAL
PUSH FILES*
*Any user with "Push" permission can upload files to Central. However, we recommend limiting who has this permission in order avoid synchronization issues.

- Security Options On your license, you can set the maximum login attempts, automatic logout settings when the system is idle, as well as password change and minimum requirements.



MadCap Central Features Per Output Type

Following are MadCap Central features supported in each output type.

	HTML5	PDF	Word	Clean XHTML	Eclipse Help	EPUB	HTML Help
Analytics	✓	✗	✗	✗	✗	✗	✗
Build Output From Central	✓	✓	✓	✓	✓	✓	✓
Private Ouptut	✓	✗	✗	✗	✗	✗	✗
Publish Directly to Central	✓	✓	✓	✓	✓	✓	✓
View Output From Central	✓	✓	✓	✓	✓	✓	✗

CHAPTER 3

Logging In and Out

If you have an account in Central, you are able to log in and out of your account from Flare. When you are logged in, you are able to access project properties, upload (bind) and import projects (via Git), and push project changes to the cloud.

> NOTE If the Central license is configured for single sign-on (SSO), you can log in through your company's identity provider. Otherwise, you would use the same password that you set in Central. For details on setting up SSO, see the Central online Help.

This chapter discusses the following:

Logging In to Central 82
Logging Out of Central 82
The SSO Login Experience 83
What's Noteworthy? 92

Logging In to Central

1. Open the project in MadCap Flare.
2. Do one of the following, depending on the part of the user interface you are using:
   - In the upper-right of Flare, click **Log In**.
   - Select **View > MadCap Central**. The MadCap Central window pane opens. Click **Log In**.
3. What happens next depends on various factors.

   If the main Central portal displays, select the **Server Location** (United States or Europe), depending on the region you chose when subscribing to central. Enter your **Email Address** and Central **Password**, then click **Log in**.

   If you are attempting to log in to a specific license, and it is set up to use SSO, you will need to enter the credentials for your company's identity provider, and maybe even use two-factor authentication (e.g., approve the login on your smart phone).

   If you are attempting to log in to a specific license, and it is not configured with SSO, enter the same password that you set up on Central previously.

   > NOTE For more details on the login experience with SSO, see the Central online Help.

Logging Out of Central

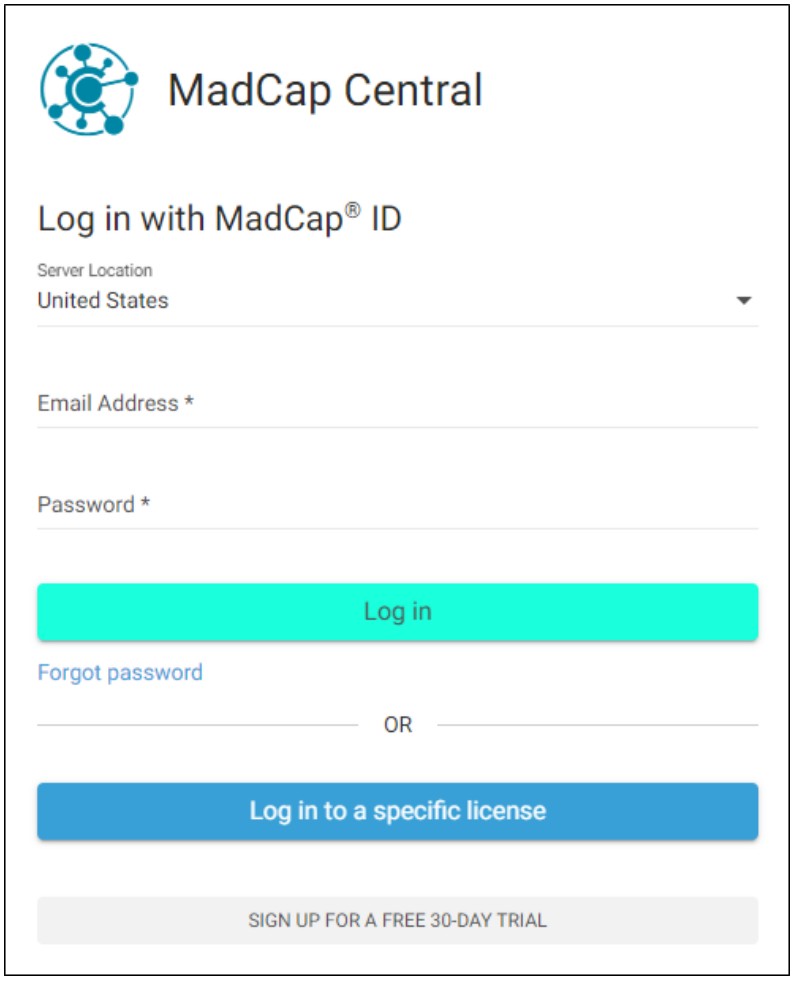
1. Open the project in MadCap Flare.
2. Do one of the following, depending on the part of the user interface you are using:
   - In the upper-right of Flare, click your avatar and select **Log Out**.
   - Select **View > MadCap Central**. The MadCap Central window pane opens. Click .

The SSO Login Experience

The single sign-on (SSO) login experience depends on different factors, such as whether it's the user's first time logging in, if there are multiple licenses, how the identity provider (IdP) is set up, etc.

First Time SSO Login

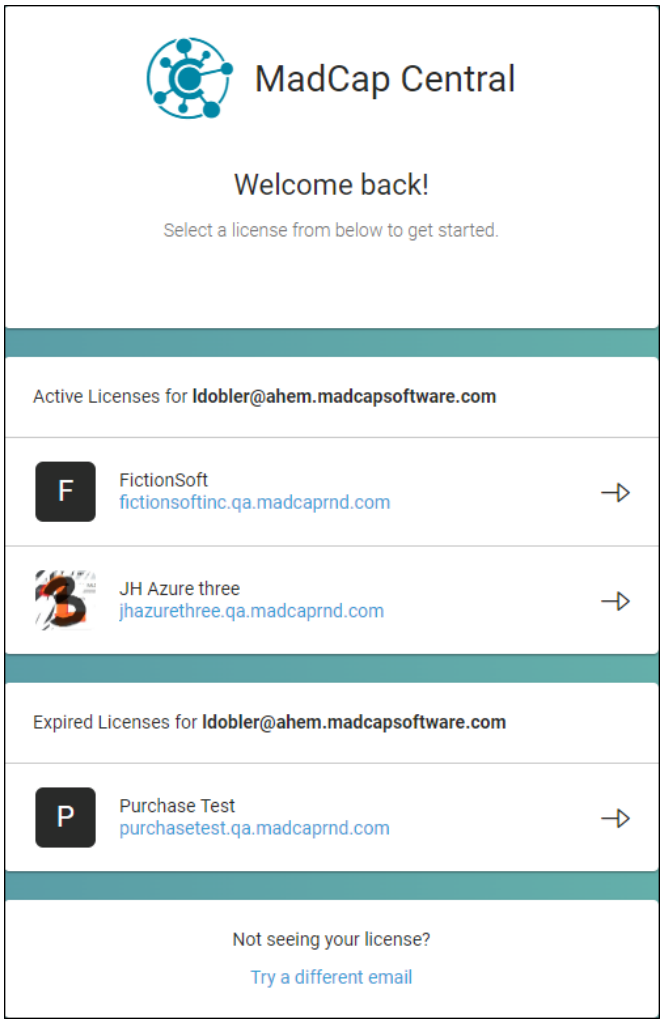
Clicking either the option in the upper-right of Flare or the option in the Central window pane opens this browser-based window if it is your first time logging in.



Select the **Server Location** (United States or Europe), depending on the region you chose when subscribing to central. Enter your **Email Address** and Central **Password**, then click **Log in**.

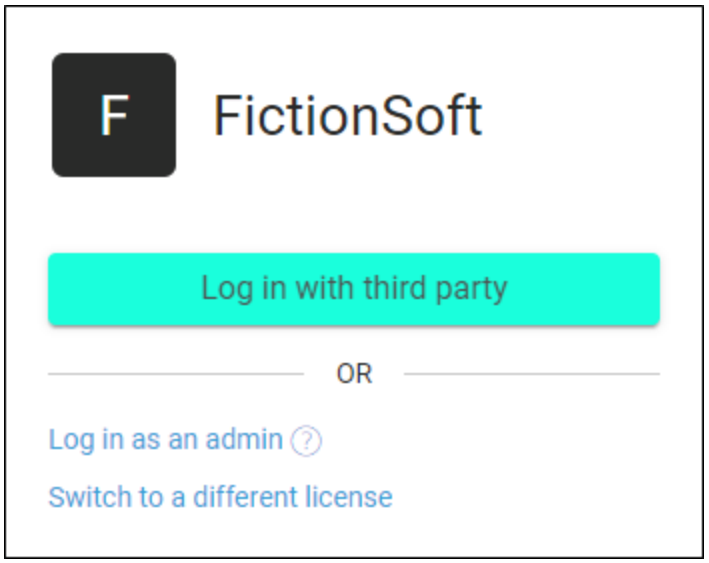
NOTE If you do not already have a Central password, you can click **Forgot password** to set one up.

Some Central users might be part of multiple Central licenses. Not only that, but some of those licenses might be enabled for SSO and some might not. Those users can click **Log in to a specific license** instead and enter the appropriate license vanity. Otherwise, clicking **Log in** displays the license hub, which shows all of your Central licenses (if you have more than one).



You can then select the Central license you want to log in to.

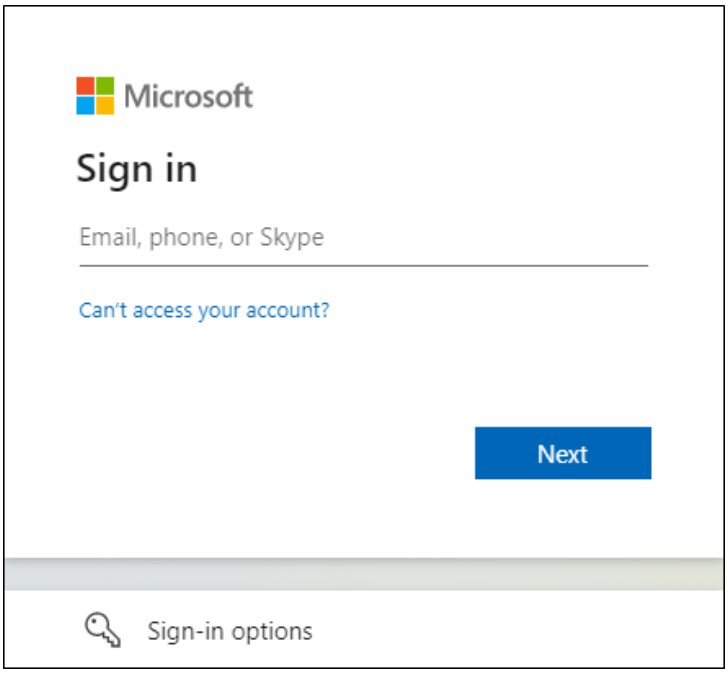
If the license is enabled for SSO, you will see the following window, although you can customize the text on the button to display something different (e.g., Microsoft Login, Okta SSO, AuthO Access).



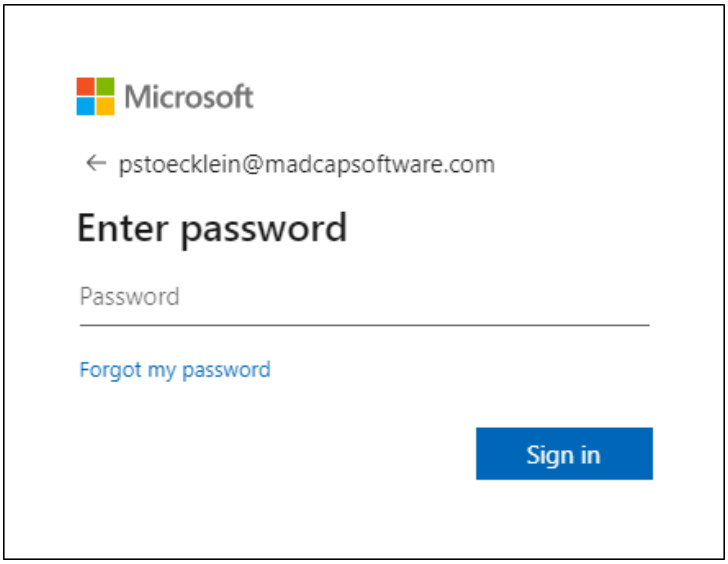
After you click the button to log in, additional windows open so you can enter credentials of some kind. The type of credentials depends on how your IT department sets up the IdP (e.g., password, verification code sent to email, two-factor authentication via smart phone).

☆ EXAMPLE — Password

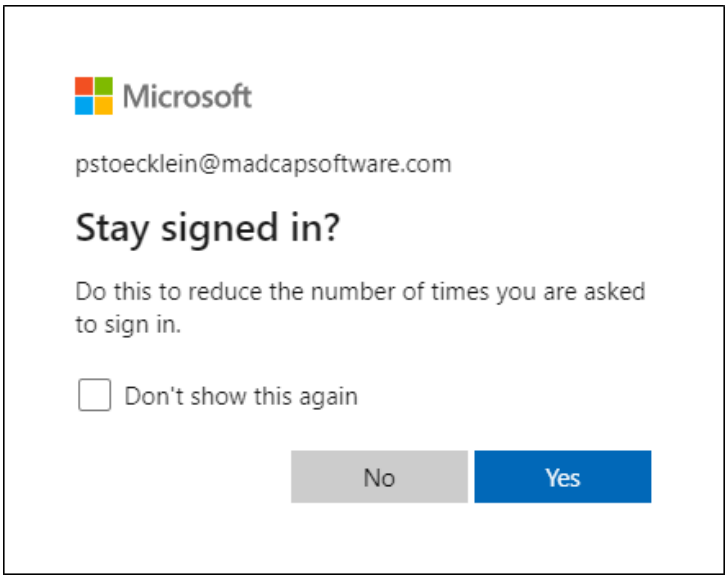
In this example, Microsoft Azure is the IdP, and it has been set up to first ask for your email, with the possibility to select other sign-in options (e.g., security key).



☆ After this, Microsoft asks for your IdP password (i.e., the password you use to log in to Windows when you start your computer).



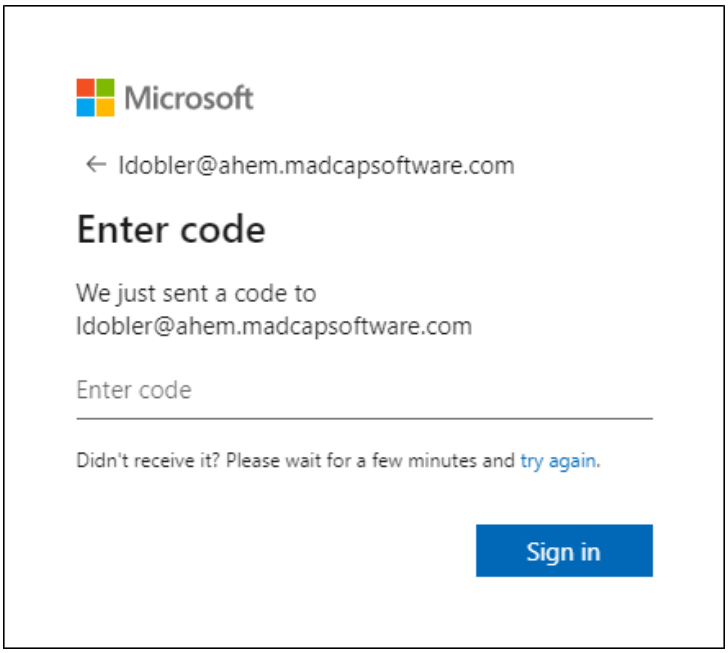
☆ Then, it might ask if you want to stay signed in.



After this window, you are logged in to Central.

EXAMPLE — Verification Code

This example is the same as the previous one, except that the IdP is set up to ask for a verification code instead of a password.

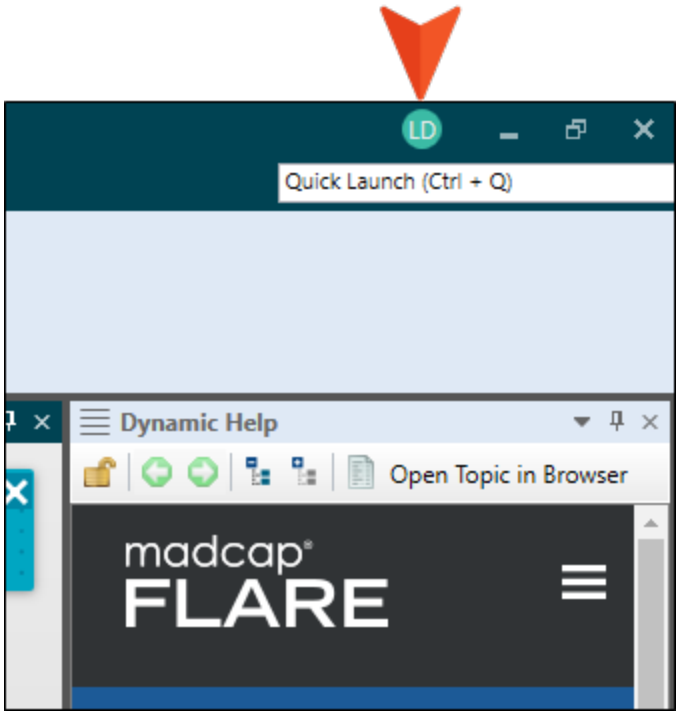


In this case, you receive an email, where the code is found.

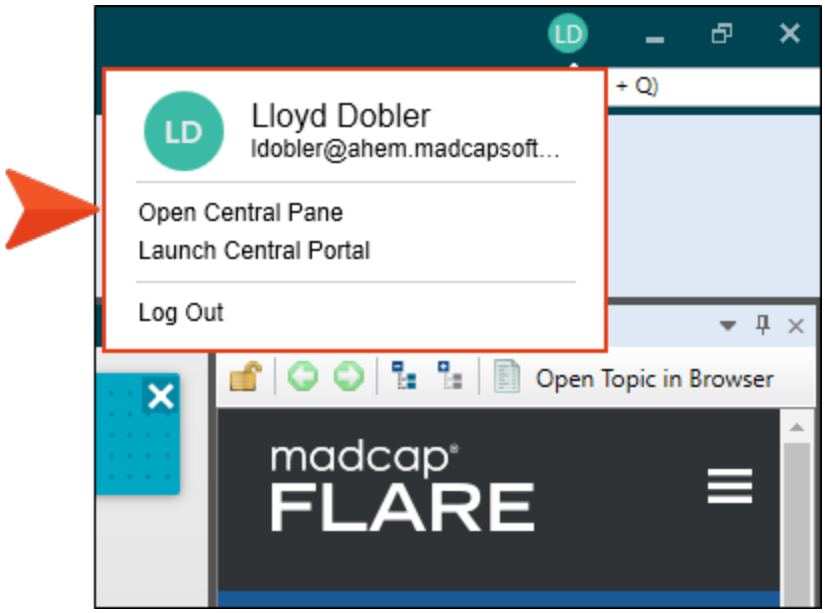
After pasting the code into the field and clicking **Sign in**, you are logged in to Central.

The Flare Interface After Logging In

After you log in to Central from Flare, the login option in the upper-right changes to display your avatar (or initials if an avatar is not yet set in Central).

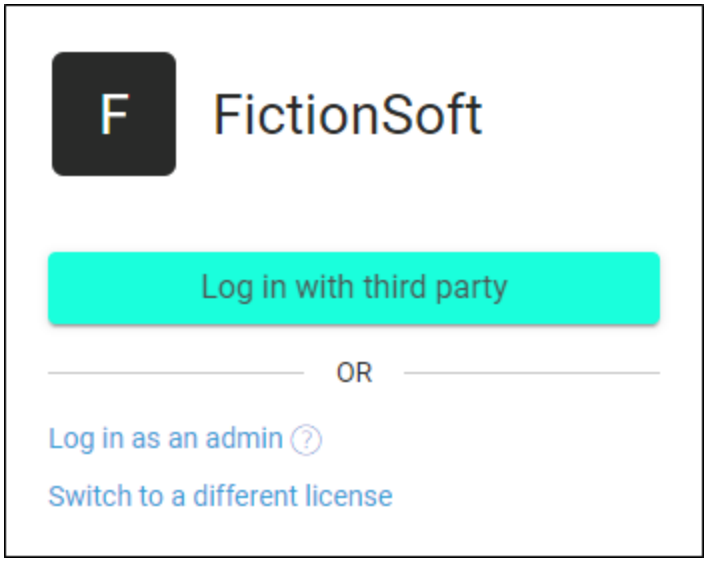


You can click the avatar to open a drop-down menu. From here you can open the Central window pane, launch the Central portal in a browser, or log out.



After the First Login

For a license that is enabled with SSO, you need to enter credentials only the first time you log in if you are using a browser. If you log out and then try to log back in, you will see the SSO login window once more.



When you click the button to log in again, you do not need to enter the IdP credentials a second time. Instead, you are simply logged in with them. This is different than a license that is not enabled for SSO, where you must enter your unique Central password each time you log in.

However, if you log out through the Flare interface (as opposed to a browser) and then log back in, it's possible you will need to enter the credentials once again.

What's Noteworthy?

NOTE Logging in or out of Central in Flare does not affect your login status in Central online. This is beneficial because you do not need to open both the Central website and Flare if you only need to work in one or the other.

CHAPTER 4

Binding a Project to MadCap Central

You can upload a project to Central from the desktop Flare application. A copy of the project files will therefore reside in the cloud via Central, and you will be able to keep the local and cloud versions of the project synchronized using integrated source control. You will also be able to generate and publish Flare targets using Central.

This chapter discusses the following:

Permission Required? 94

How to Upload (Bind) a Project to Central 95

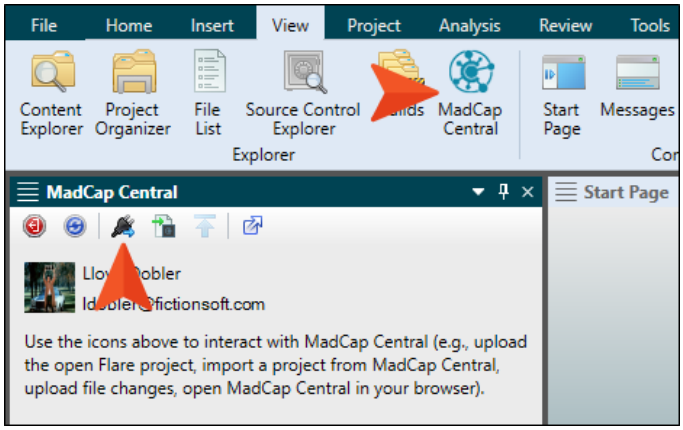
Permission Required?

For this activity, you must have the following permission setting in Central:

For more information about permissions, see the Central online Help.

How to Upload (Bind) a Project to Central

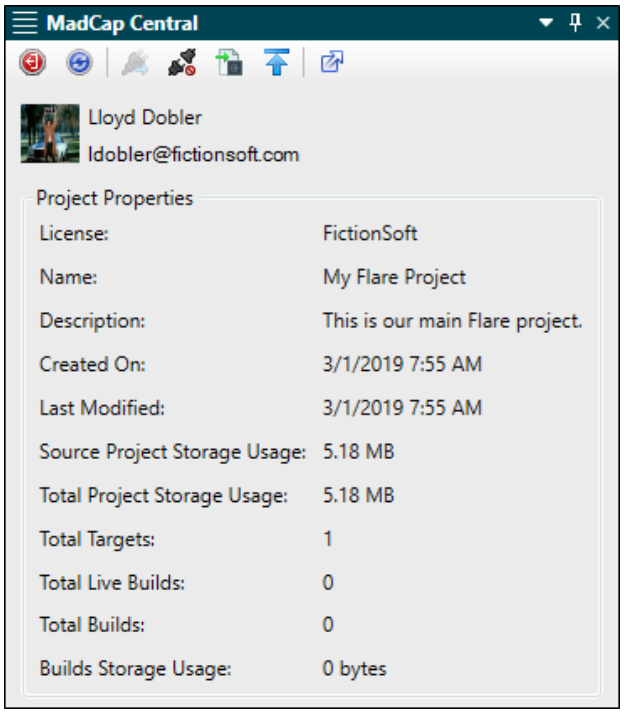
1. Open the project in MadCap Flare.
2. Select **View > MadCap Central**. The MadCap Central window pane opens.
3. Be sure to log in if you aren't already.
4. Click .



> **TIP** If this button is disabled, try clicking in the window pane first.

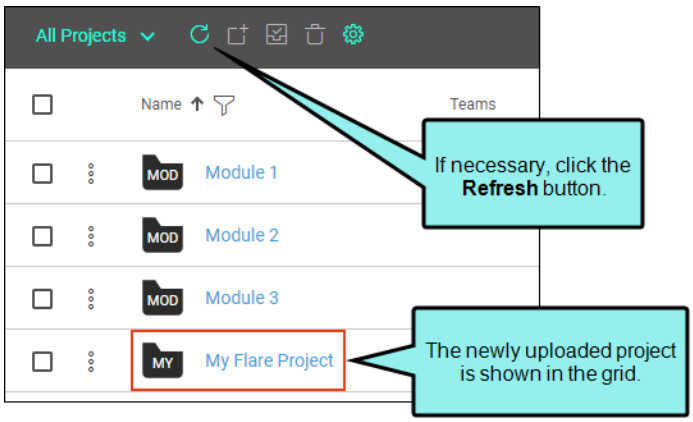
5. In the Bind Project dialog, you can complete these fields, then click **OK**.
   - **Enter project name** This shows the name of the project. You can change the name if you want, but in most cases you will probably want to keep the same name.
   - **Enter project description** You can enter a short description, which can then be seen when anyone views the project profile in Central.

6. In the success message, click **OK**. The MadCap Central window pane adjusts, showing properties for the current project.



7. (Optional) In the local toolbar, click to open Central.

8. (Optional) In Central, you can click **Projects** on the left side to open the Projects page. The uploaded project should be shown in the grid. If it isn't, click to refresh the page.



NOTE If your project is dual-bound (see "Dual-Bound Projects" on page 16), be sure you check in or push the project files to your third-party source control provider after you bind to Central. This will ensure that other users who import the project from source control have the most recent version of the project.

NOTE When you bind a project to Central, a .gitignore text file is created in your local project folder. Advanced users can edit the .gitignore file with a text editor to control which files or folders can be pushed to your Git repository. You can also specify which files and folders are ignored by Git, and are not pushed to the repository.

NOTE If you intend to build a target that links to an outside Flare project (e.g., for the purpose of importing files via Global Project Linking, project merging, or multilingual output), make sure that the other project is also uploaded to Central. Otherwise, the automated imports or merging will not work.

NOTE Binding and unbinding is usually straightforward when using the Flare interface. However, if you bind to projects outside of Flare, there is more you should know.

NOTE Due to security issues, MadCap Software does not support the use of the Secure Shell (SSH) protocol in MadCap Flare when connecting to MadCap Central. The Hypertext Transfer Protocol Secure (HTTPS) protocol is supported.

CHAPTER 5

Importing Projects From Central

If another user uploads a Flare project to your MadCap Central license and you do not yet have that project on your computer, you can import it.

In order to import a project, you must be associated with that project in Central. For more information, see the Central online Help.

IMPORTANT If you are using a Git-Central dual-binding, you can import projects from Git (using the Import Project From Source Control Wizard) or from Central. The project will be the same. However, if you import from Central, you will only be able to push changes to and pull changes from Central. If you import from Git, you will be able to push changes to and pull changes from Central *as well as* your main Git repository. This is because the project in Central has no connection to the original Git repository, and if you import from Central you will not have those source control bindings.

NOTE If your project is already bound to a third-party source control provider other than Git (i.e., you are working in a dual-bound model as opposed to a single-bound model—see "Dual-Bound Projects" on page 16), the Import option in the MadCap Central window pane will be disabled. This is because you should import your project directly from source control using your non-Central binding.

If you already have the project on your machine, you do not need to reimport the project from source control. Instead, you can get the latest version from source control by using the Flare Source Control ribbon (see the Flare online Help).

Be sure that you have the latest version of the project.

NOTE When a project is dual bound to Central and a non-Git third-party provider, keep the following in mind if you (i.e., the second user) want to access this setup. After importing the project from Perforce Helix Core, Subversion, or Team Foundation Server, you will have to re-bind the project to Central. In Flare's MadCap Central window pane, click to upload (or bind) the project to Central. In the Bind Project dialog, it is important to enter the exact name of the project as it currently exists on Central. When you click **OK**, a message displays asking if you want to bind to the existing project. Select **Yes**. This re-establishes the existing Flare/Central connection; note that it does not create a new binding.

This chapter discusses the following:

Permission Required? 100
How to Import a Project from Central 101

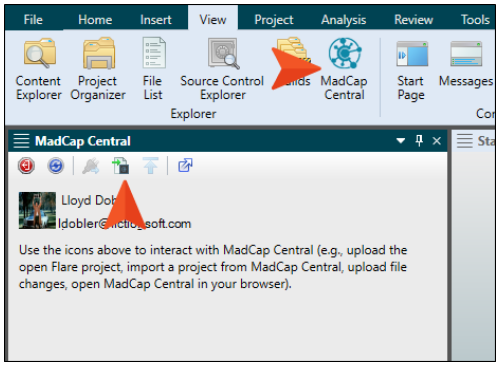
Permission Required?

For this activity, you must have the following permission setting in Central:

For more information about permissions, see the Central online Help.

How to Import a Project from Central

1. Select **View > MadCap Central**. The MadCap Central window pane opens.
2. Be sure to log in if you aren't already.
3. Click .



> **TIP** If this button is disabled, try clicking in the window pane first.

The Import Project dialog opens.

4. From the **My Projects** field, select the project you want to import.
5. Next to the **Destination folder** field, click and choose the folder where you want the imported project to be stored on your computer.
6. Click **OK**.
7. When the import is finished, you can click **Open** to launch the project in Flare. Otherwise, you can click **Cancel**.

CHAPTER 6

Viewing Project Properties

If you have access to a project in Central (i.e., you have been added to the project), you can view its properties in the MadCap Central window pane. This lets you see relevant information about the project, such as when it was added to Central, the amount of space it is using on Central, and the number of targets and builds.

How to View Project Properties in Flare

1. Open the project in MadCap Flare.
2. Open the MadCap Central window pane (View > MadCap Central).
3. If you are not already logged in to Central, log in to your account.

   Once you are logged in, the project's properties appear in the MadCap Central window pane. You can view the following properties:

   - License The name of the license where the project was uploaded.
   - Name The name of the project.
   - Description The project description (if applicable).
   - Created On The date the project was added to Central.
   - Last Modified The last time you pushed changes to the project to Central.
   - Source Project Storage Usage The total size of the project's source files.
   - Total Project Storage Usage The total size of the project's files (source plus output).
   - Total Targets The number of targets in the project.
   - Total Live Builds The number of builds for this project that are currently set to "Live" in Central (i.e., the builds are published so that the public can see the output).
   - Total Builds The number of builds for this project.
   - Builds Storage Usage The amount of storage in Central used by builds.

CHAPTER 7

Source Control Activities

Some activities are particularly common and important when it comes to this feature.

This chapter discusses the following:

Committing and Synchronizing (Pull, Push) in a Single-Bound Model 105

Pushing in a Dual-Bound Model 110

Committing and Synchronizing (Pull, Push) in a Single-Bound Model

If you are working in a single-bound model (see "Single-Bound Projects" on page 12), you author content in Flare. When you want to transfer your changes to Central, you must commit and synchronize the changes.

Permission Required?

For this activity, you must have the following permission settings in Central:

For more information about permissions, see the Central online Help.

Commit

To commit a changed file is to record it to the local repository. Essentially you are saying, “I’m ready to transfer this content up to the cloned project on Central.” Committing files gives you the opportunity to organize them into different groups when you add them to Central. You can also add a unique comment to each commit.

☆ EXAMPLE You’ve made changes to 23 files in your Flare project. Maybe 17 of the files are related to Feature A that you are documenting and the other 6 are related to Feature B.

Suppose your company policy is that you must add a comment each time you upload changes and provide a summary of what you did. To keep the summary for Feature A separate from Feature B, you decide to do two commits.

First, you use the Pending Changes window pane in Flare to select the 17 changed files related to Feature A. Then you perform the commit and add a relevant comment.

After this, you select the 6 changed files related to Feature B and perform another commit with a different comment.

How to Commit Files in a Single-Bound Project

1. Open the project in MadCap Flare and make your changes.

   > NOTE If you are using Git branching, make sure the appropriate branch is active. When you commit the changes, it will be for that branch.

2. Do one of the following, depending on the part of the user interface you are using:

   - Status Bar In the lower-right of Flare, click (a number indicates how many files contain changes that need to be committed).

     > NOTE If you do not see this option, make sure **View > Status Bar** is enabled.

   - Pending Changes Window Pane From the **Source Control** ribbon, open the Pending Changes window pane. Select the files in the window pane that you want to commit, and in the local toolbar click .
   - Ribbon Select **Source Control > Commit** (for selected files) or **Source Control > Commit All** (for all files in the project).
   - Right-Click If you have the Content Explorer, Project Organizer, Pending Changes window pane, or File List open, right-click the file you want to commit and select **Source Control > Commit** (for selected files) or **Source Control > Project > Commit All** (for all files in the project).

   The Commit dialog opens. The selected files are listed with check boxes next to them.

3. Enter a comment tied to the commit. This enables you to keep an audit trail for a file. The comment can then be viewed from the History dialog, which can be accessed from the Source Control Explorer, the Source Control ribbon, the File menu, or the Source Control button .

4. Click **Commit**.

Synchronize

To synchronize means to pull, then push, committed changes between the cloned Central project and your local Flare project. This allows you to retrieve changes that other writers have uploaded to Central from their local projects, and they can get your changes as well. You cannot synchronize changes from Central, but you can from Flare.

How to Synchronize Files in a Single-Bound Project

It is possible to use the Pull and Push options in the Flare interface individually to synchronize your files, but it is more convenient to use the Synchronize option instead. Flare will first pull changes from the project on Central. After that, it will push changed files from the local project up to Central.

1. Do one of the following in the Flare project, depending on the part of the user interface you are using.

   NOTE If you are using Git branching, make sure the appropriate branch is active. When you synchronize files, it will be for that branch.

   - **Status Bar** In the lower-right of Flare, click (a number indicates how many commits need to be pushed or pulled).

     NOTE If you do not see this option, make sure **View > Status Bar** is enabled.

   - **Ribbon** Select **Source Control > Synchronize**.
   - **Right-Click** If you have the Content Explorer, Project Organizer, Pending Changes window pane, or File List open, right-click any file and select **Source Control > Project > Synchronize**.

2. (Optional) If you did not commit your files before starting the synchronize, a dialog asks if you want to commit your files. Click **Yes** to continue.

   NOTE You must commit *all* modified files to proceed with the synchronization.

3. In the Select Remote for Synchronize dialog, select the remote repository (if necessary) and click OK. If you are using a dual-bound setup, origin is typically the name of the repository for the first binding, and MadCapCentral is the name for the Central binding.

   If no conflicts were discovered during the synchronization, you do not need to continue with the following steps; you are finished.

   If conflicts were found (i.e., a remote file is different from the version in your local repository), the Resolve Conflicts dialog opens. Continue with the following steps.

4. Do one of the following:

   - If you want to accept all of the differences between the remote and local files, thus merging them, click Auto Merge All. If this step is a success, you do not need to continue with these steps.
   - If you want to review the differences in the files side by side and resolve each conflict (or if auto-merging is not possible due to conflicts occurring in the same location in a file), click Resolve. The Resolve Version Conflict dialog opens.

5. From the Resolve Version Conflict dialog you can choose from the following options:

   - Merge changes in merge tool Opens a merging interface, which lets you see exactly what changes were made and choose which to keep.
   - Undo my local changes Automatically removes your changes and keeps changes from other authors.
   - Discard external changes Automatically removes changes from other authors and keeps your changes.

6. If you selected the option to use the merge tool, the Merge Changes dialog opens. Use this dialog to view and select changes. You can take actions in the following ways.
    - Click a Change Use the key at the top of the dialog, as well as the color coding on the local and server sides, to determine if a change has been added (new), deleted, changed, moved, or is in conflict (difference occur in the same paragraph). For conflicts in the same paragraph (i.e., areas where a diagonal line is shown), you can click the icon next to either the local or server change and choose Keep Change. This will copy that change to the text area at the bottom of the dialog.
    - Type Content If you want to use your changes as well as those from another author, and even tweak the paragraph a bit more, you can click in the area at the bottom of the dialog and simply type content.
    - Previous/Next Conflict When you are finished resolving the first conflict, you can use the "Previous Conflict" and "Next Conflict" buttons at the bottom of the dialog to work on other conflicts in the file.

    NOTE If you selected "Merge as Text" in the local toolbar and are working in the code, you can click on text with a hatched background to keep the change in it. After you click on text with a hatched background, the hatched lines are removed, leaving a solid color.

7. After all conflicts have been resolved, click OK. A message lets you know that a backup of the file has been created in case you need to roll back to it. Click OK.
8. Click OK in any remaining dialogs.
9. Because you encountered conflicts, your changes were not pushed up to Central. Therefore, click the Synchronize option again to complete the process.

Pushing in a Dual-Bound Model

If you are working in a dual-bound model (see "Dual-Bound Projects" on page 16), you will author your content in Flare and then use your third-party source control provider to synchronize your files with those from other writers. After this, you can use Flare to push those changes to Central.

> NOTE It is possible for multiple people working on a dual-bound project to push files to Central. However, if you are using a source control provider other than Git for the first binding, the most recently pushed files are the ones that are used in Central. In other words, the last person to push "wins." To avoid issues, be sure that you have the most recent version of file changes from all other writers in your local project before you push. You may even want to limit users' permissions (see the Central online Help) so only one or two people are allowed to push files to Central.

Permission Required?

For this activity, you must have the following permission setting in Central:

For more information about permissions, see the Central online Help.

How to Push Files in a Dual-Bound Project

1. Open the project in MadCap Flare.
2. Get the latest version of the files from your third-party source control provider. For more information, see the Flare online Help.
3. Select **View > MadCap Central**. The MadCap Central window pane opens.
4. Be sure to log in if you aren't already.
5. In the local toolbar of the MadCap Central window pane, click .
6. In the dialog to select the remote repository, make sure **MadCapCentral** is selected. If it is not, click the drop-down and choose it. Then click **OK**.

   A progress dialog shows you the status of the push. If the changes are pushed successfully, a confirmation will appear.

> NOTE When you are dual-bound using Git-Central (see "Dual-Bound Projects" on page 16), you can push to both Central and Git. It is possible that the files in the main Git repository and the files in Central will become out-of-sync if changes are made in one repository (i.e., Git or Central) and not made in the other. If this happens, you will see the Flare Conflict Resolution dialog. You can use this dialog to accept or reject other users' changes. For more information about merging source control files and conflict resolution, see the Flare online Help.

CHAPTER 8

Analytics Activities

Some activities are particularly common and important when it comes to this feature.

This chapter discusses the following:

Creating and Editing Analytics Keys 113
Associating Analytics Keys With Targets 114
Managing Analytics Keys 115
Downloading Analytics 116

Creating and Editing Analytics Keys

Analytics keys are associated with Flare HTML5 targets. You can create and edit these keys either in Flare or on Central. To do this, you must have the permission on Central to "Manage Output Analytics Keys" (see the Central online Help).

How to Create or Edit an Analytics Key in Flare

1. Open an HTML5 target.
2. Select the **Analytics** tab.
3. From the **Provider** field, select **Central**.
4. If you are not yet logged in to Central, click **Login**. Then complete the fields in the dialog and click **OK**.
5. From the **Central License** field, choose your license.
6. Next to the **Key** field, click **Create**.

   If you already have a key that you want to edit, select it from the drop-down, then click **Edit**.
7. Enter a name for the key and an optional description, then click **OK**.
8. Click to save your work.

NOTE After creating or editing an analytics key, and associating it with a Flare target, you need to build the target and publish the output for the analytics to take effect and see results on Central.

Associating Analytics Keys With Targets

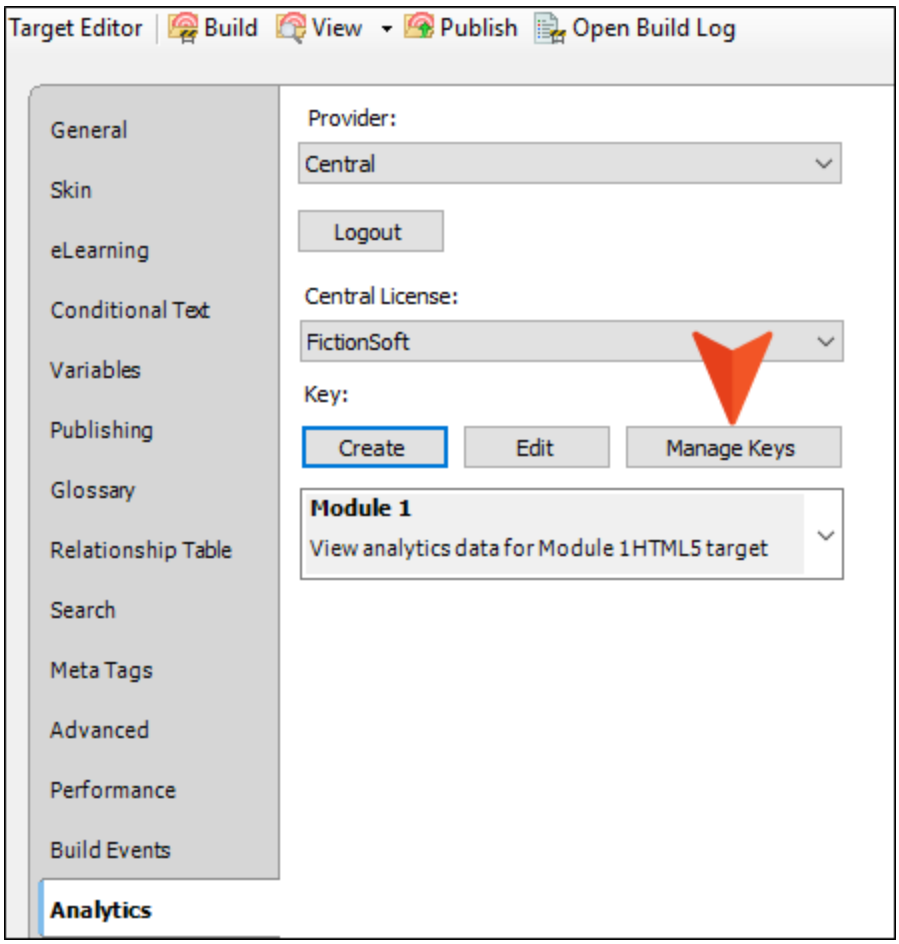
If you create a key in Flare, it will become associated with your target as you do this. But if you create the key in Central, you need to then open the Flare target and associate that key with the target.

How to Associate an Analytics Key With a Target

1. Open an HTML5 target.
2. Select the **Analytics** tab.
3. From the **Provider** field, select **Central**.
4. If you are not yet logged in to Central, click **Login**. Then complete the fields in the dialog and click **OK**.
5. From the **Central License** field, choose your license.
6. At the bottom of the **Key** area, click the drop-down and select the name of the key.
7. Click to save your work.
8. Build output for the target on Central to see the analytics results.

Managing Analytics Keys

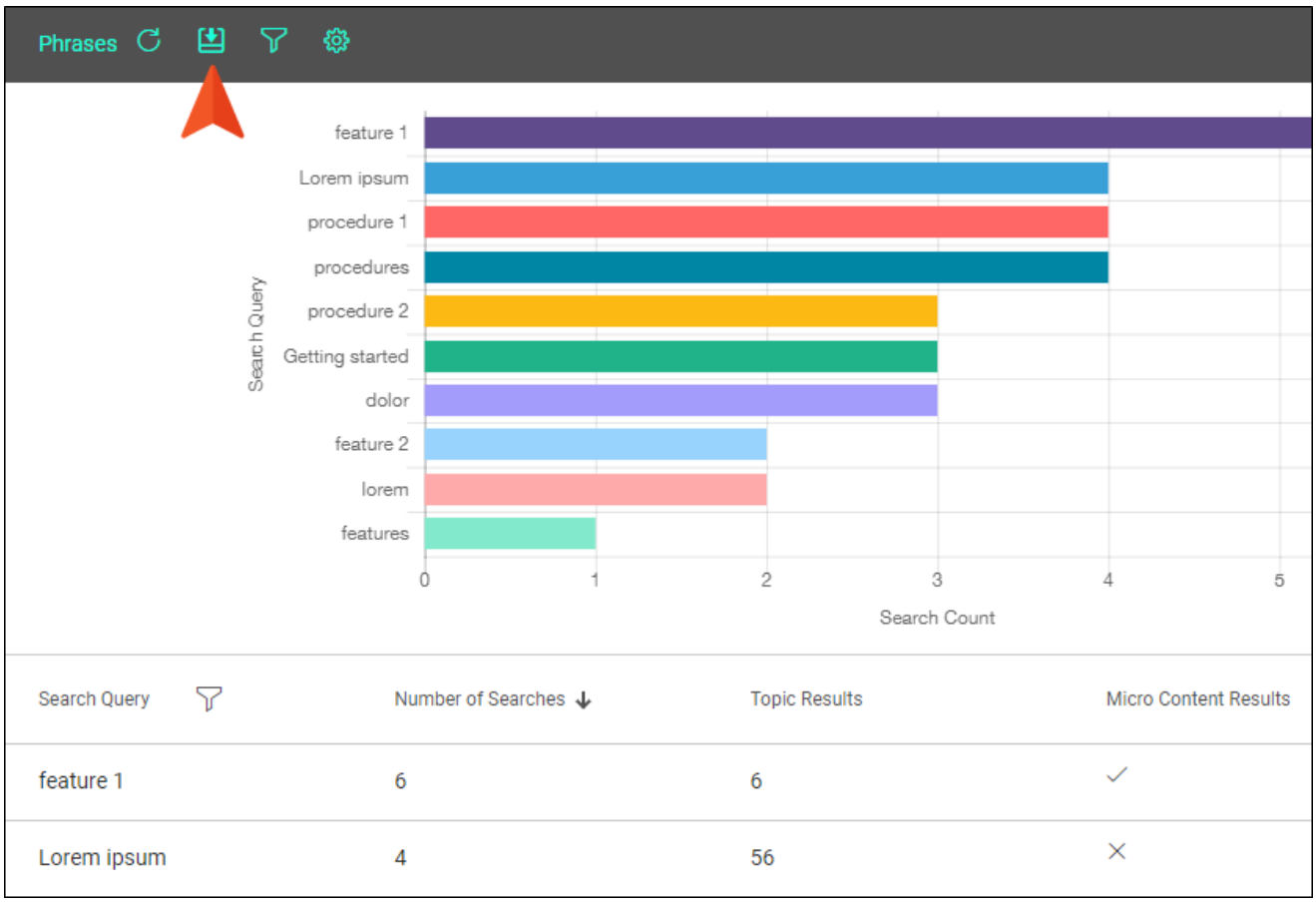
You can click Manage Keys in the Flare target.



This opens the Analytics page in Central. From here, you can create and edit keys, as well as delete any that you no longer need.

Downloading Analytics

In the toolbar of the Analytics page, you can click to download the data to a comma-separated value (CSV) file.

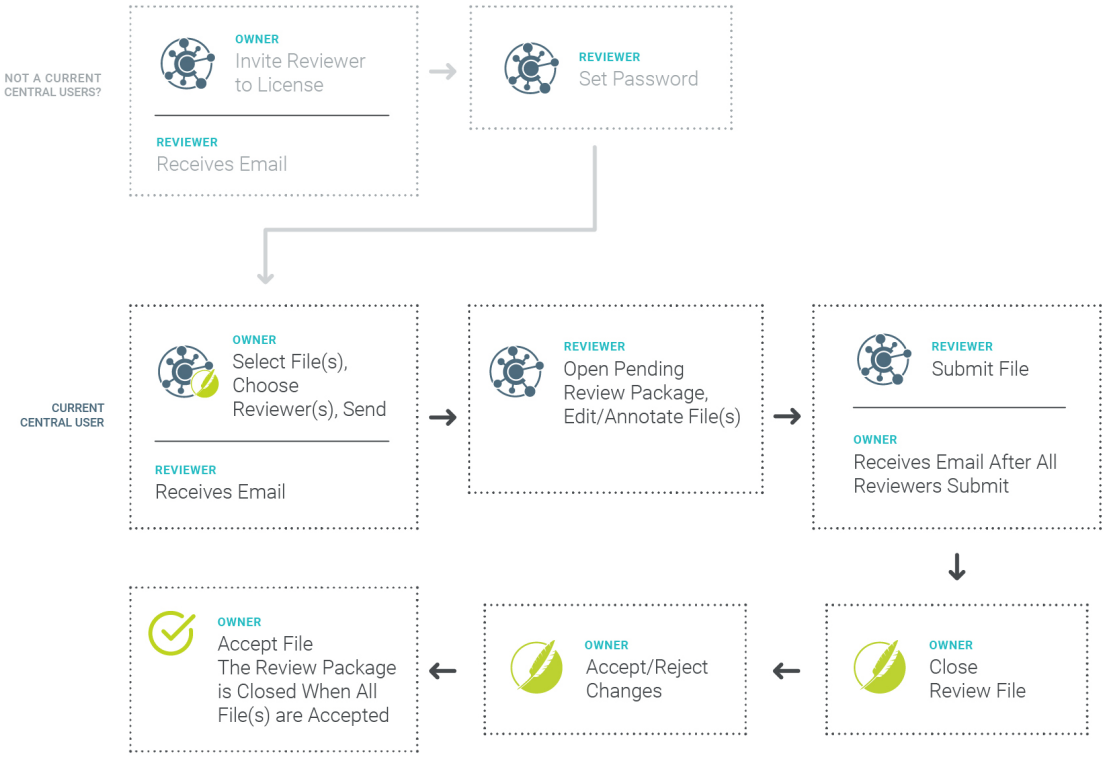


Search Query	Number of Searches	Topic Results	Micro Content Results
feature 1	6	6	✓
Lorem ipsum	4	56	✗

CHAPTER 9

Central Review Process

Flare supports review packages (i.e., bundling files such as topics and snippets) and lets you send those packages to Central to be reviewed by others. After editing and annotating the files in Central's Review Editor, the reviewers submit the finished files, sending them back to your inbox in Flare. You can then accept or reject their changes and accept the file, replacing the original source file.

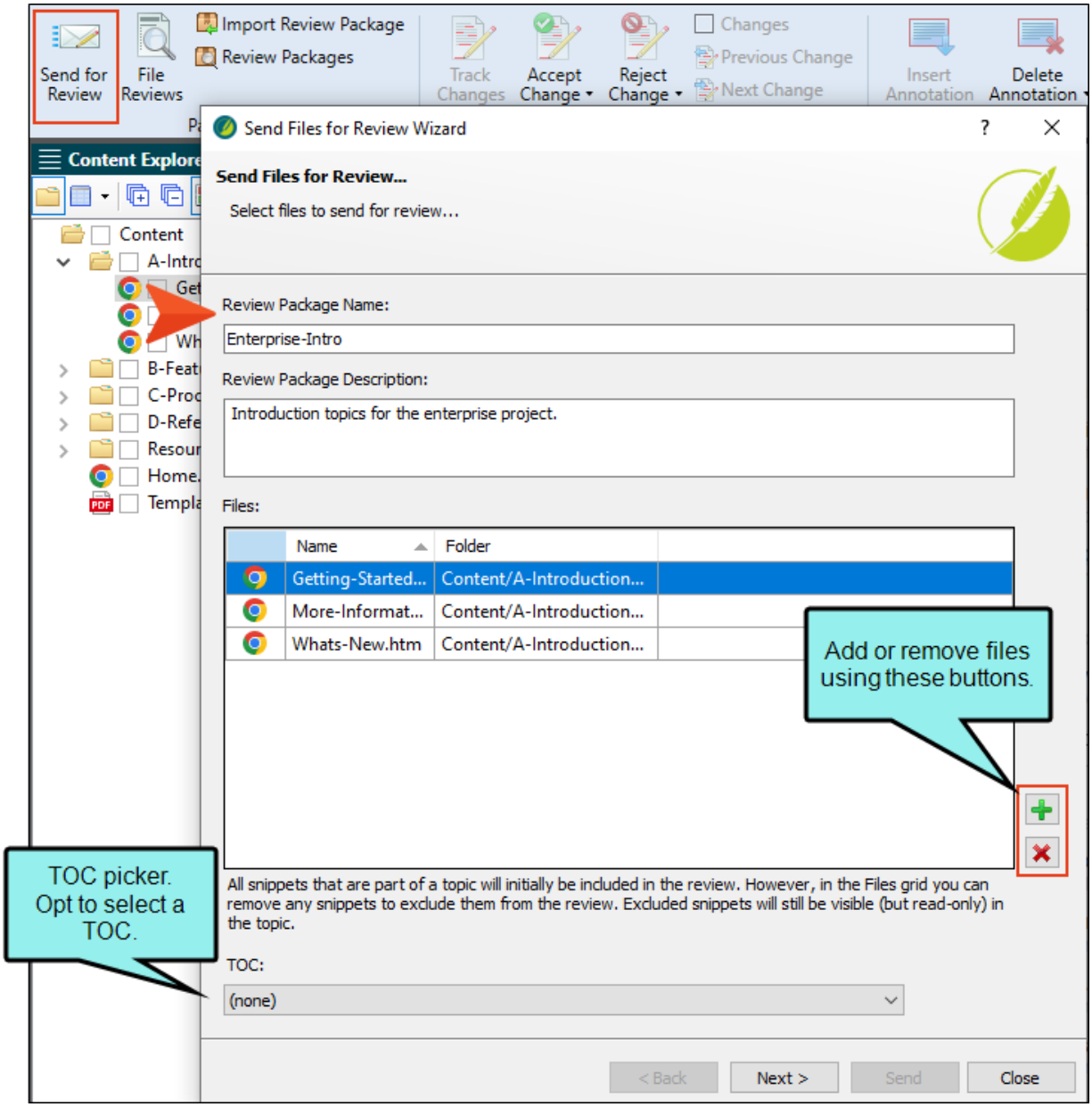


This chapter discusses the following:

Workflow 119
Benefits 130
What's Noteworthy? 131
Sending Files for Review on Central 132
Closing Central Review Files 139
Accepting or Rejecting Tracked Changes (Central Reviews) 142
Accepting Reviewed Files 146

Workflow

1. **Owner** In Central, open the **Users** page and invite the reviewer(s) to Central.
2. **Reviewer** Click the link in the email to accept the invitation and set a password.
3. **Owner** In Flare, switch to the appropriate branch (if necessary), select **Review > Send For Review**. The first screen of the Send Files for Review Wizard opens.
    a. In the **Review Package Name** field, enter a name.
    b. (Optional) Enter a description for the review package.
    c. (Optional) Add or remove files for the package.
    d. (Optional) The TOC drop-down defaults to **(none)**. To associate files within a review package to a TOC in the project, select a TOC.
    e. Click **Next**.

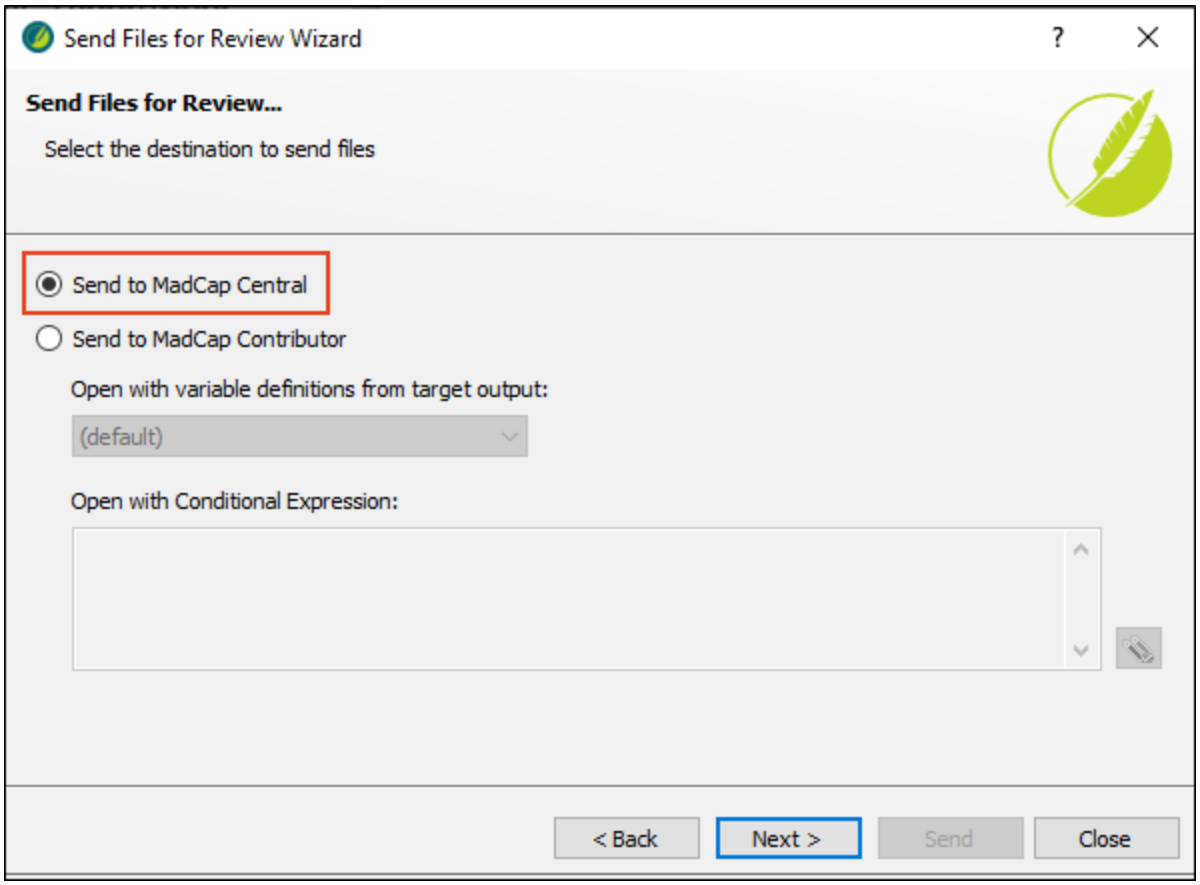

Send for Review
File Reviews
Import Review Package
Review Packages
Track Changes
Accept Change
Reject Change
Changes
Previous Change
Next Change
Insert Annotation
Delete Annotation
Content Explore
Content
Send Files for Review Wizard
Send Files for Review...
Select files to send for review...
Review Package Name:
Enterprise-Intro
Review Package Description:
Introduction topics for the enterprise project.
Files:
Name
Folder
Getting-Started...
Content/A-Introduction...
More-Informat...
Content/A-Introduction...
Whats-New.htm
Content/A-Introduction...
Add or remove files using these buttons.
TOC picker. Opt to select a TOC.
All snippets that are part of a topic will initially be included in the review. However, in the Files grid you can remove any snippets to exclude them from the review. Excluded snippets will still be visible (but read-only) in the topic.
TOC:
(none)
< Back
Next >
Send
Close

TIP All snippets that are part of a topic will initially be included in the review. However, when sending files for review from Flare, you can remove any snippets from the File grid to exclude them from the review. You might choose to exclude some (or all) snippets from the review if you think it will be too confusing and too many files for your reviewers. If you exclude snippets, they will still be visible in the topic in Central, but the snippets will be read-only. Therefore, reviewers won't be able to make edits in the snippets, but in the topic they can insert annotations (comments) around the snippets in order to provide feedback.

NOTE If you are using Git branching, make sure the appropriate branch is active. When you send files for review, they will be associated with that branch.

In Central, reviewers can determine which branch a file is from. Reviewers do not need to do anything special to make edits or comments in the files.

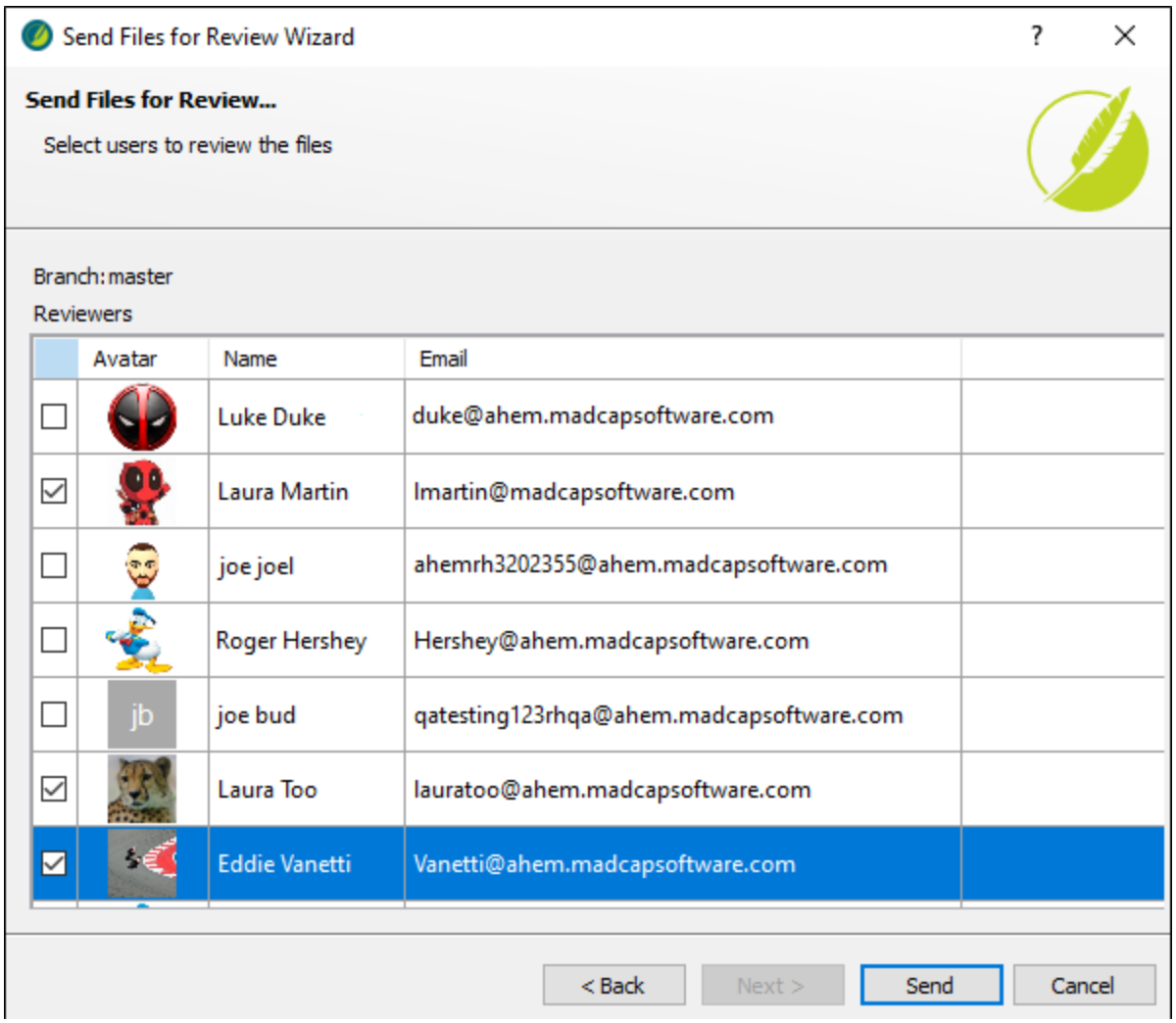
4. **Owner** Select **Send to MadCap Central** (if not already selected), and click **Next**.



NOTE If necessary, Flare also prompts you to commit and synchronize any changes in your project with Central before you can proceed to the next page of the wizard.

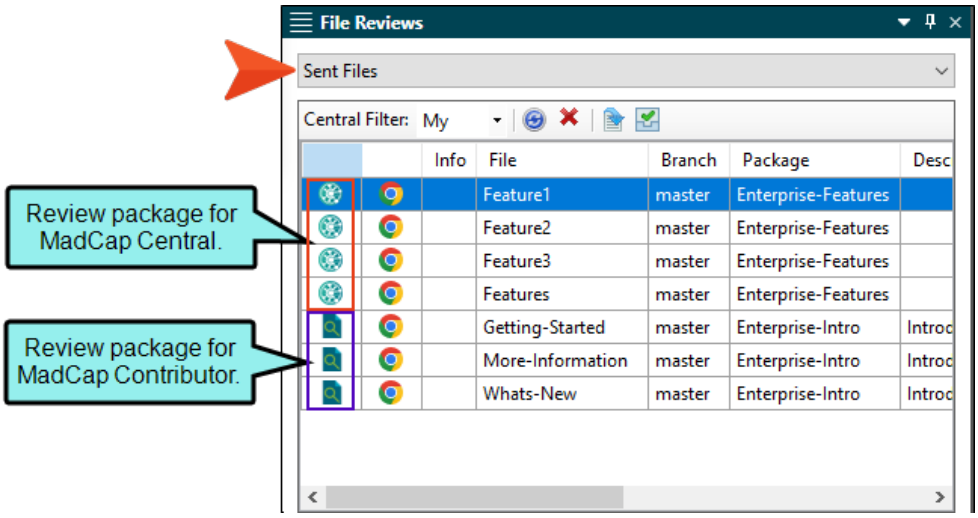
5. Owner (Optional) Select reviewers for the review package, and click Send. The wizard closes and a prompt alerts you that the files have been sent to Central, click OK.

   The owner can choose reviewers of different seat types, such as authors and SMEs. Anyone selected during this process automatically becomes associated with the Flare project on Central, and becomes an assigned reviewer for all files chosen in the wizard.



After the review package is uploaded, reviewers receive an email notification with a link to open the Reviews page in Central. (SMEs automatically receive email notifications, but Flare authors must have the review notifications and email option enabled in Central to receive emails.)

In File Reviews, the review package files display in the Sent Files area. This means the files are in an "Open" state, pending review.



NOTE If you attempt to edit a file that has been sent out for review and is still in an "Open" state, a warning displays. You can edit the file, but you may have conflicts with the version of the file being reviewed. As an alternative to editing the file in Flare, you can edit it in Central along with your reviewers until it is finished.

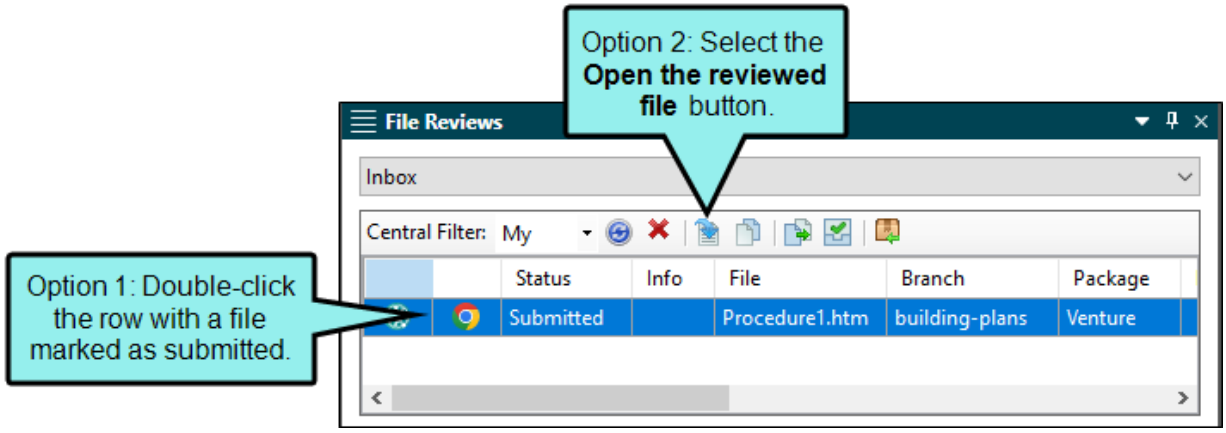
6. Reviewer In Central, open the Reviews page. Click each file associated with the pending review package and edit or annotate it using the Review Editor, and click Submit.

   The reviewer is done submitting once all the files in the queue are complete. This moves the review package to the Submitted pane. Files can be viewed from there and edited still, but once the owner closes the file, it can no longer be edited in Central.

   After all of the reviewers submit the file(s), the owner receives an email notification (if notifications are enabled in Central) and the file(s) displays in the File Reviews Inbox in Flare.

NOTE If the owner of a review package makes edits to the files in Central they will not typically see a Submit button.

7. **Owner** After reviewed files are individually submitted from Central, open Flare. Select **Review > File Reviews**. You can look at files as they come in; you do not have to wait for all the files in a review package in order to preview them.
8. **Owner** From the File Reviews Inbox area double-click a submitted file. (Or, select a file, and from the local toolbar, click **Open the reviewed file** .)

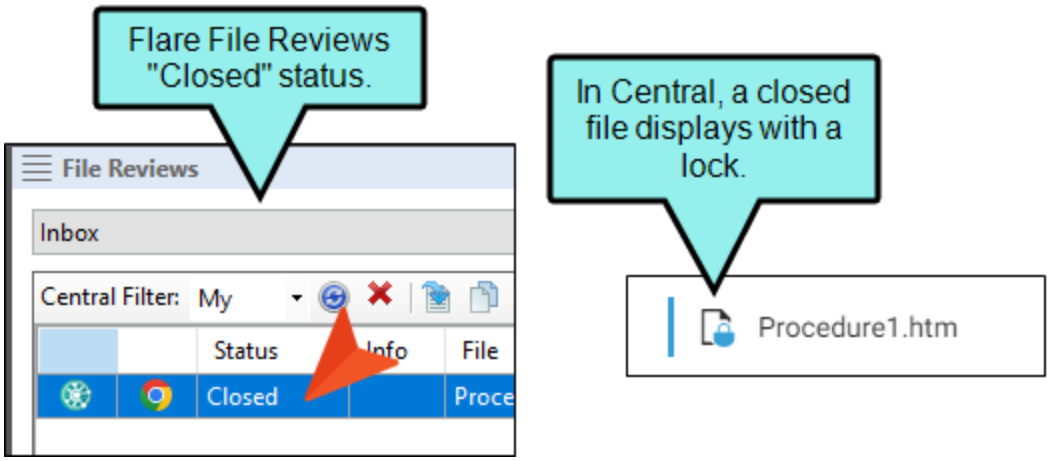


NOTE The Inbox displays files from all relevant branches (if you are using Git branching). You should switch to the matching branch before managing the tracked changes and accepting the file back into Flare.

NOTE Any authors working in that Flare project can close the review file. This is necessary in case the owner (i.e., original author) is not available to close it.

9. Owner When you click to open and preview a file, a dialog displays asking if you want to move the files to a closed state. Select Yes or No.

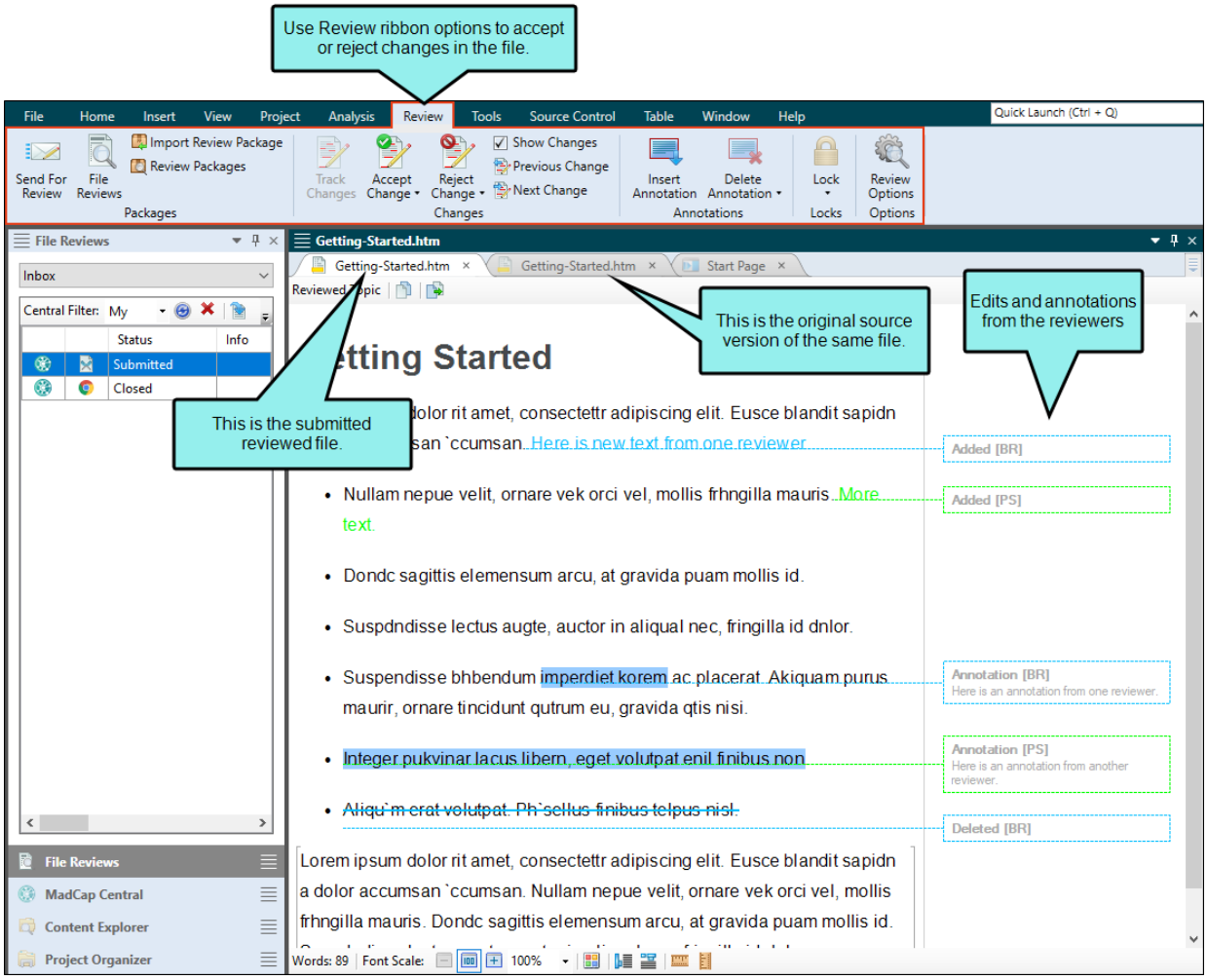
    - Yes The file switches to a closed state in the Inbox and opens in the editor. No more editing can occur in Central while the owner reviews it. In Central, when a file is closed, a lock symbol appears next to the file.



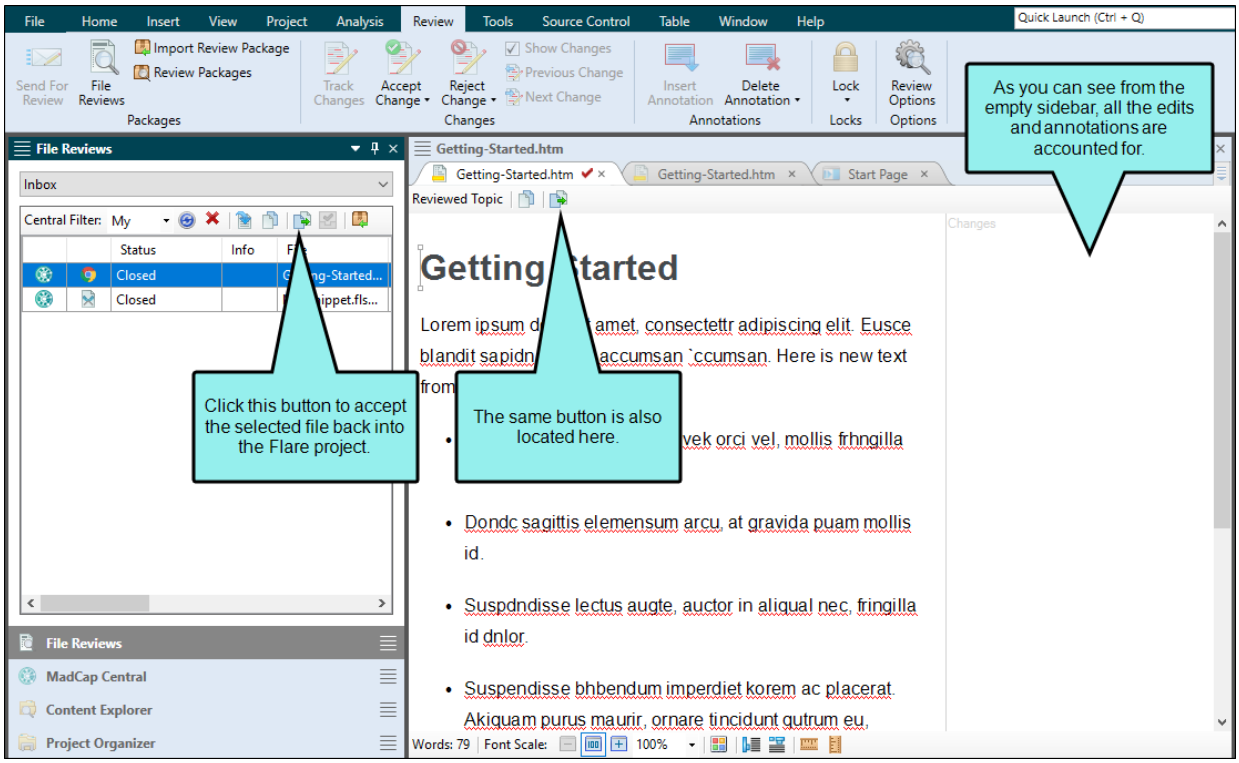
    - No The owner can preview the file in Flare—as a read-only file. The reviewer(s) can still make changes to the file in Central until the owner closes the file in Flare. The file stays in a submitted state in the Inbox, and it opens in the editor.

> NOTE The advantage of this is that you can see raw edits before the reviewer is done—without having to close the file and resend it from Flare. When you are ready to close out the file from further editing, double-click the submitted file and select Yes from the prompt.

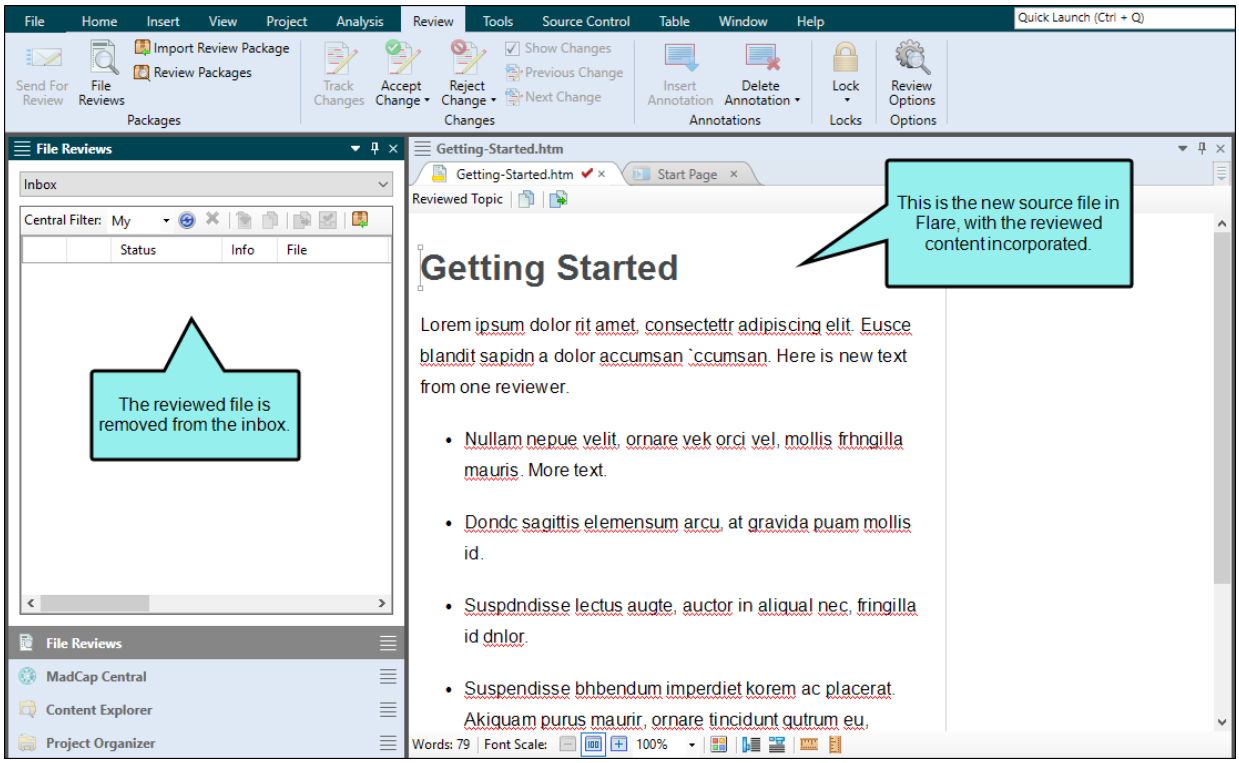
10. **Owner** In Flare, accept and/or reject the reviewer's changes in the editor.



11. **Owner** In the toolbar in Flare, click the **Accept** button.



The file is then removed from the Inbox.



Benefits

Benefits of this review workflow include:

- Cloud Review SMEs do not need to download and install any software. The review takes place in the cloud.
- Multi-User Editing Multiple reviewers (e.g., SMEs, authors) can make changes and add comments to the same topic or snippet at the same time.
- Review-Only Interface A lightweight version of the editor means a streamlined interface. You only see options and features that are relevant to the review process.
- Auto-Save and Tracking Changes in the editor are auto-saved as you work. In addition, all changes are automatically tracked so the owner can easily locate edits for approval or rejection.
- Branching Files can be sent for review from a specific Git branch. This lets you keep reviews limited to files that are still in a state of development, as opposed to finished and ready for publication.
- Workflow Management Bundling files into review packages favorably services everyone in the review process workflow. Reviewers (e.g., SMEs) can edit and view content from a flat file list or from a contextual TOC view where files display in a TOC for output that an end user might see. Owners (e.g., authors) can edit but they can also monitor and manage reviews in Central. They have access to customizable package and file grids that show review information and progress.

What’s Noteworthy?

NOTE If you are using Git branching, you can send review packages from a particular branch; this way, reviewers can look at content that is in a state of development, as opposed to content that is considered completed and published.

NOTE For the review process, keep in mind that review packages are sent up, and files come down. In other words, owners are sending review packages up to Central, and then reviewers are submitting individual files back down to Flare. Once all package files are accepted into Flare, the review package is closed.

Sending Files for Review on Central

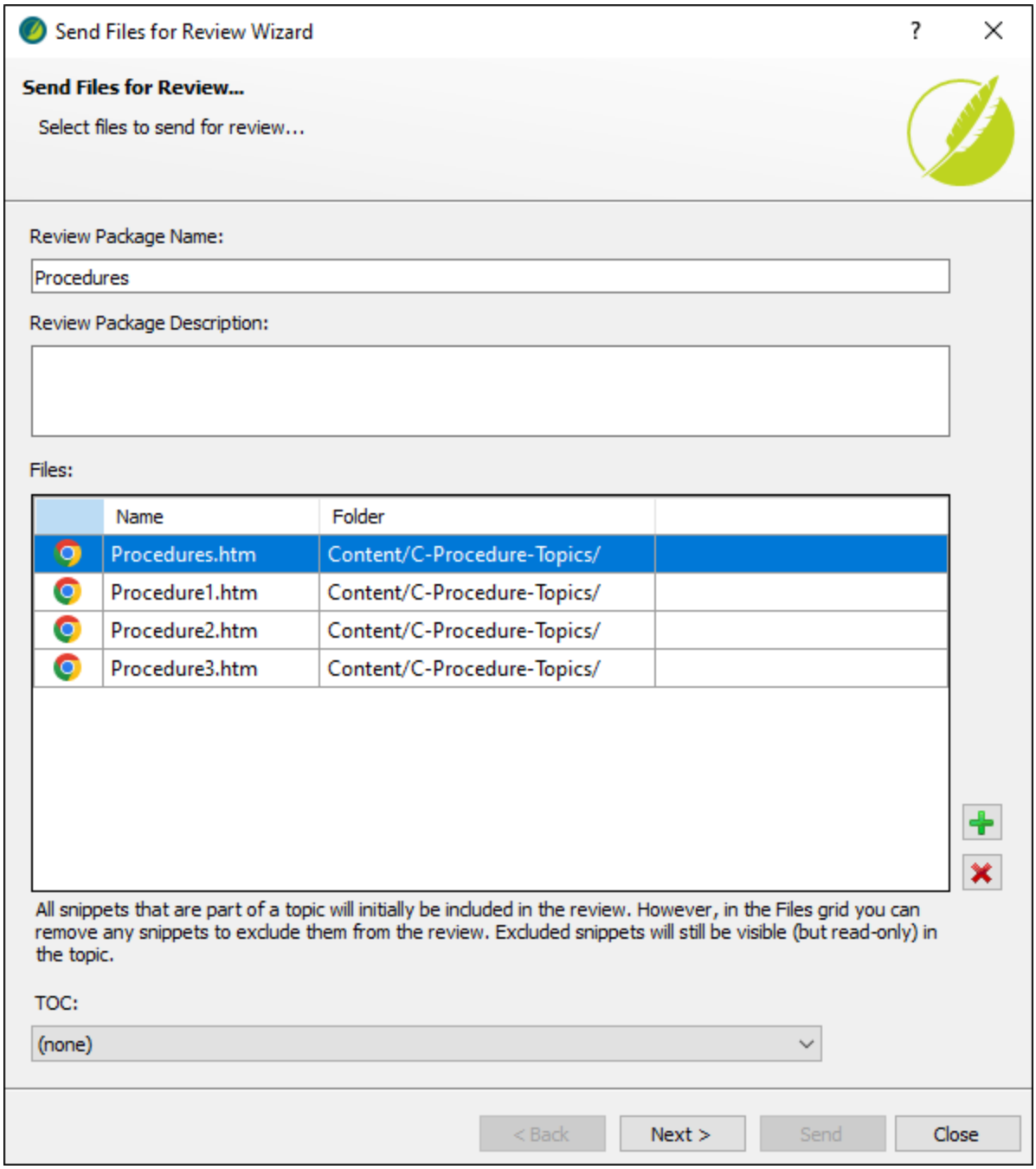
The following are steps for sending files to others for review in Central.

How to Send Files for Review on Central

1. If you have not already done so, open the MadCap Central window pane in Flare (**View > MadCap Central**) and log in to Central. See "Logging In and Out" on page 81.
2. If you are using Git branching, make sure the appropriate branch is active. When you send files for review, they will be associated with that branch.
3. From the Content Explorer, select a file for review (e.g., topics, snippets). (From the local toolbar, you can **Toggle Show Files** to select multiple files at once.)

> NOTE You can also send files for review using the TOC Editor. The advantage to this is that the TOC node is automatically bundled with the files. This allows reviewers in Central to see content in a TOC view for better context. From the TOC Editor, select the folder(s) or file(s), right-click and select **Send for Review**. (Alternatively, from the TOCs local toolbar you can select **Send for Review**.)

4. Select **Review > Send for Review**. The Send Files for Review Wizard opens.



5. (Optional) If you want to add more files to the package, do the following:
    a. Click .
    b. Use the multi-view buttons to find the file(s) you want to add. If you want to add multiple files, make sure is selected.

	Shows or hides the folders that the files are stored in.
	Shows or hides the files. If you click this button when the Show Folders button is selected, the area splits into two halves. The folder is shown on the left side, and the files and subfolders within it are shown on the right.
	Shows or hides hidden files and folders.
	If the Show Files button is the only one selected, you can click this button to move up one folder level.

    c. Find and select the file(s) you want to add to the review. You can hold the SHIFT key to select a range, or you can hold the CTRL key to select individual items. When you are finished click **Open** in the dialog.

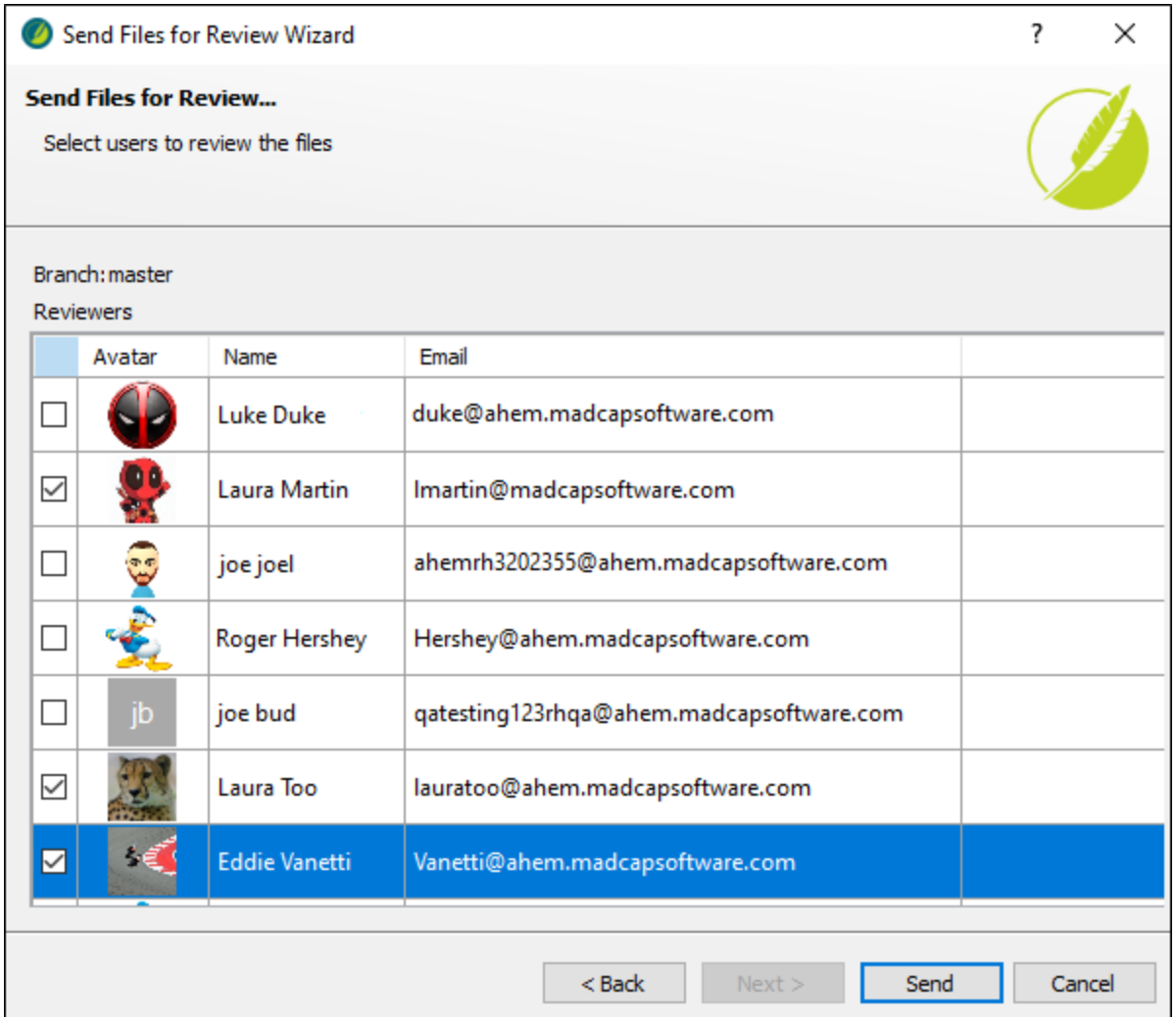
> TIP All snippets that are part of a topic will initially be included in the review. However, when sending files for review from Flare, you can remove any snippets from the File grid to exclude them from the review. You might choose to exclude some (or all) snippets from the review if you think it will be too confusing and too many files for your reviewers. If you exclude snippets, they will still be visible in the topic in Central, but the snippets will be read-only. Therefore, reviewers won't be able to make edits in the snippets, but in the topic they can insert annotations (comments) around the snippets in order to provide feedback.

6. (Optional) When sending files from the ribbon, the Send Files for Review Wizard TOC field defaults to (none). You can still associate a TOC with the review files. From the **TOC** drop-down, select a TOC.
7. Click **Next**.

8. Select **Send to MadCap Central**. Click **Next**.

> NOTE If necessary, Flare also prompts you to commit and synchronize any changes in your project with Central before you can proceed to the next page of the wizard.

9. (Optional) Select reviewers on your Central license who you want to review the package. Anyone selected automatically becomes associated with that Flare project on Central, and becomes an assigned reviewer for all files chosen in the wizard.

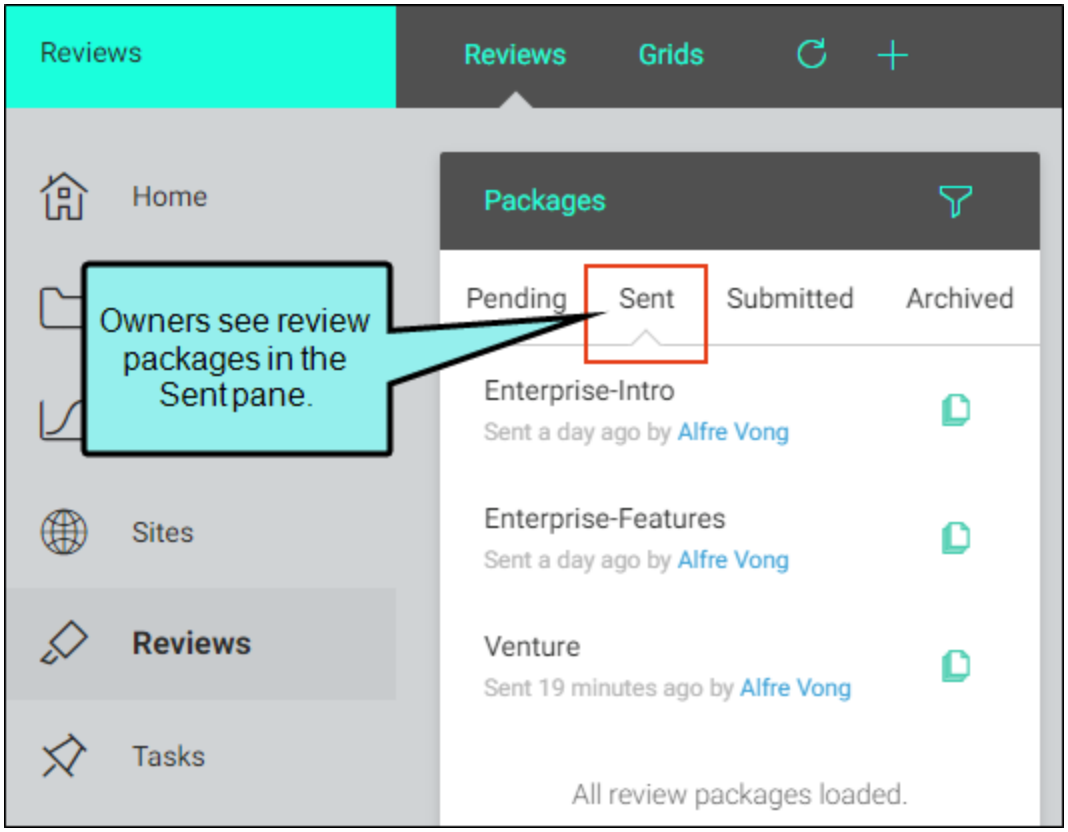


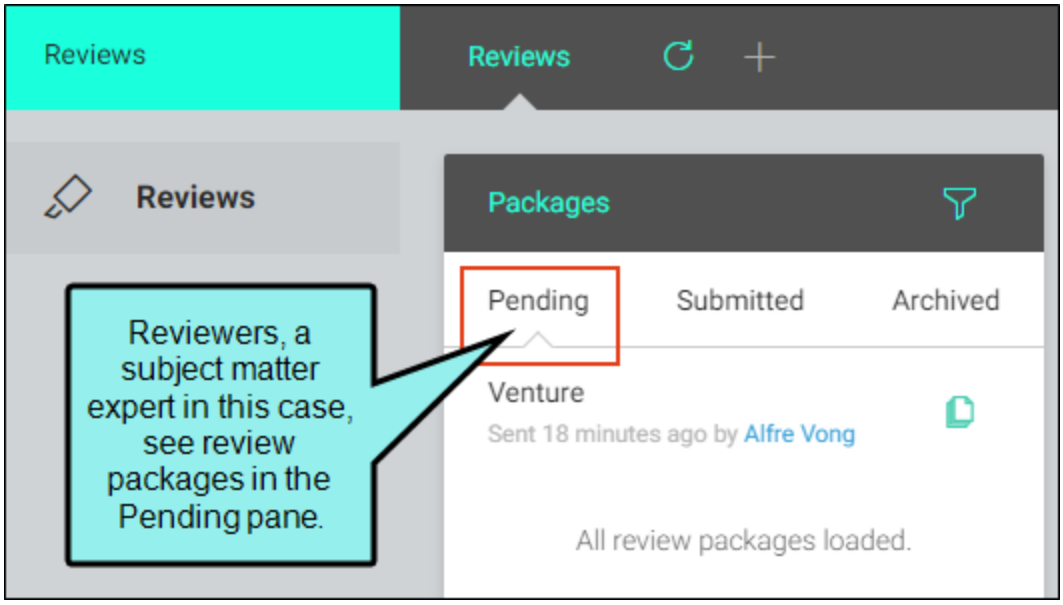
NOTE You can select users who are active on the Central license, and those who are still in the "Invited" status (i.e., those users have not yet set up a profile to become active users). After invited users are active, they will receive review notifications and see all files sent to them for review.

10. Click Send.

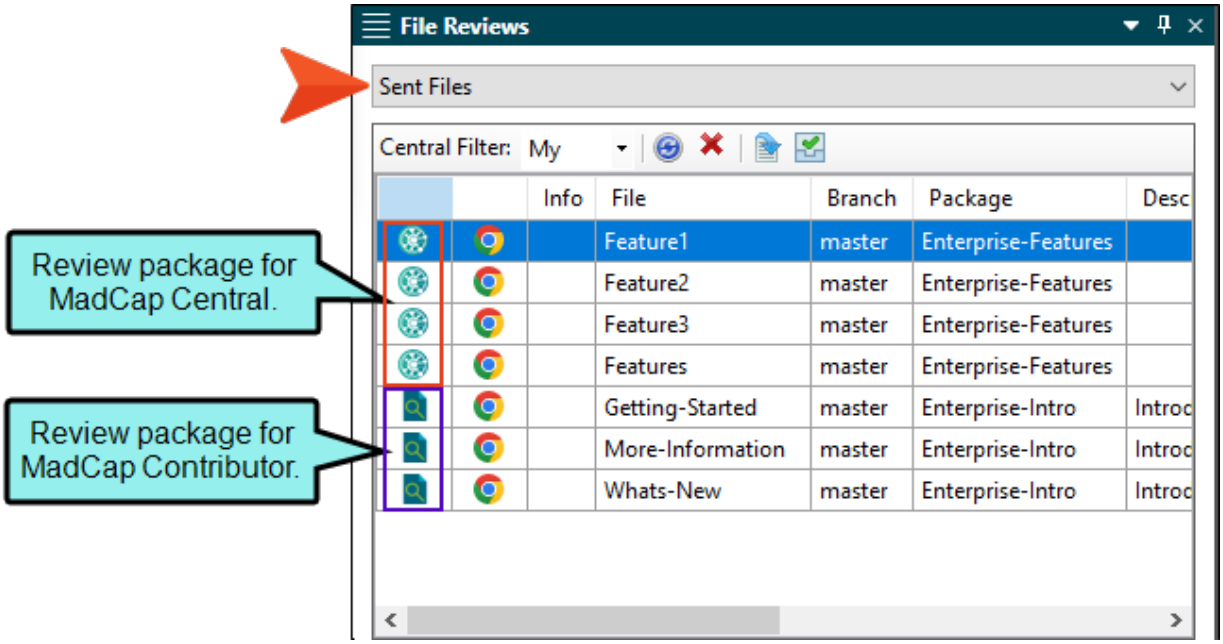
Central Side After the review package is uploaded, reviewers receive an email notification with a link to open the Reviews page in Central. (SMEs automatically receive email notifications, but Flare authors must have the review notifications and email option enabled in Central to receive emails.)

If you are the owner, review packages are listed in the Sent pane. If you are a reviewer, the review packages are listed in the Pending pane.





Flare Side In File Reviews, the review package files display in the Sent Files area. This means the files are in an "Open" state, pending review.



NOTE If you attempt to edit a file that has been sent out for review and is still in an “Open” state, a warning displays. You can edit the file, but you may have conflicts with the version of the file being reviewed. As an alternative to editing the file in Flare, you can edit it in Central along with your reviewers until it is finished.

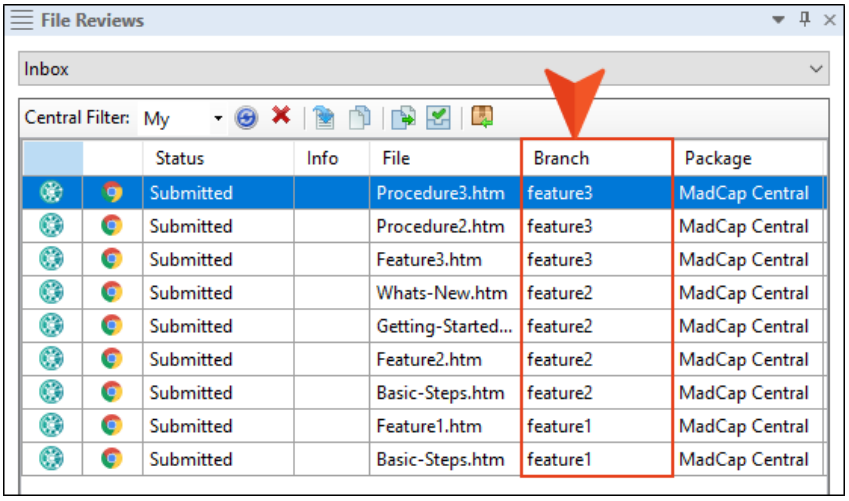
Closing Central Review Files

After all reviewers submit a file in Central, the author can close out the review process for that file in Flare's File Reviews window pane.

How to Close the Review Process for Files

1. Select **Review > File Reviews**. The File Reviews window pane opens.

> NOTE If you are using Git branching, make sure the appropriate branch is active. When closing the review process for files, dealing with tracked changes, and accepting the file back into the project, you want to be working in the correct branch so that the file changes are not applied in the incorrect branch.
>
> In the File Reviews window pane, you can see which branch a file originates from.



2. From the drop-down at the top of the window pane, select **Inbox**. The files that need to be closed have a status of Submitted. The Inbox shows files that have not been accepted.

   > NOTE Technically, you can select **Sent Files** in the drop-down and close files from there. However, these are files that reviewers have not yet finished in Central. So if you close the review process for these files, they can no longer be edited on Central. You might close the review process for a file from Sent Files, for example, if a particular reviewer leaves the company and is unable to finish editing the file on Central.

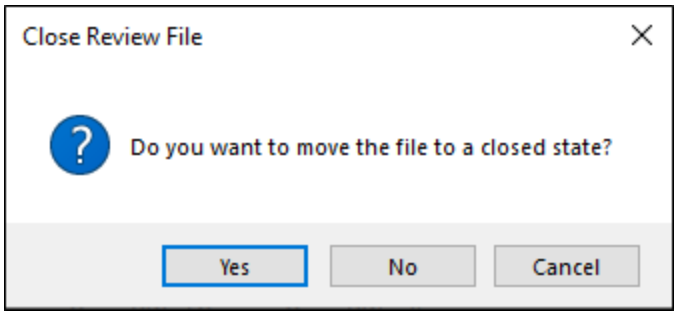
3. (Optional) By default, the **Central Field** field in the local toolbar is set to **My**, which means you see only files that you have initiated the review process. You can change the setting to **All** if you need to see files initiated by other authors in the Flare project. This option is useful if the original author is unavailable to close out the review process.

4. Do one of the following:

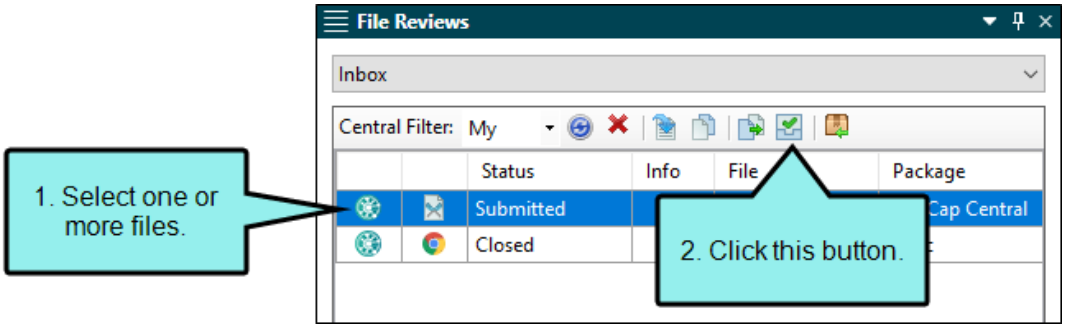
TO CLOSE REVIEW PROCESS FOR A SINGLE FILE

a. Do one of the following:

- Double-click a submitted file. A dialog displays asking if you want to move the files to a closed state. Select **Yes**.



- Select the file in the grid and click in the local toolbar. In the dialog that displays, click **OK**.



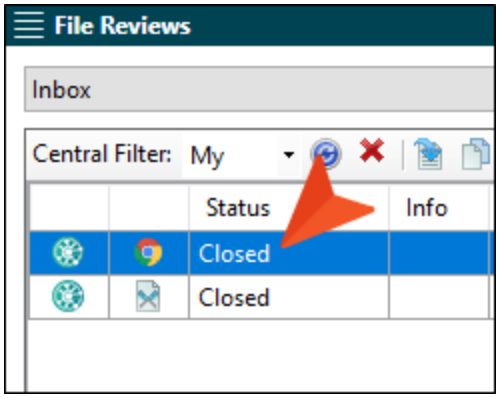
TO CLOSE REVIEW PROCESS FOR A MULTIPLE FILES

a. In the grid, select the files. You can hold the **SHIFT** key to select a range, or you can hold the **CTRL** key to select individual items.

b. In the local toolbar, click .

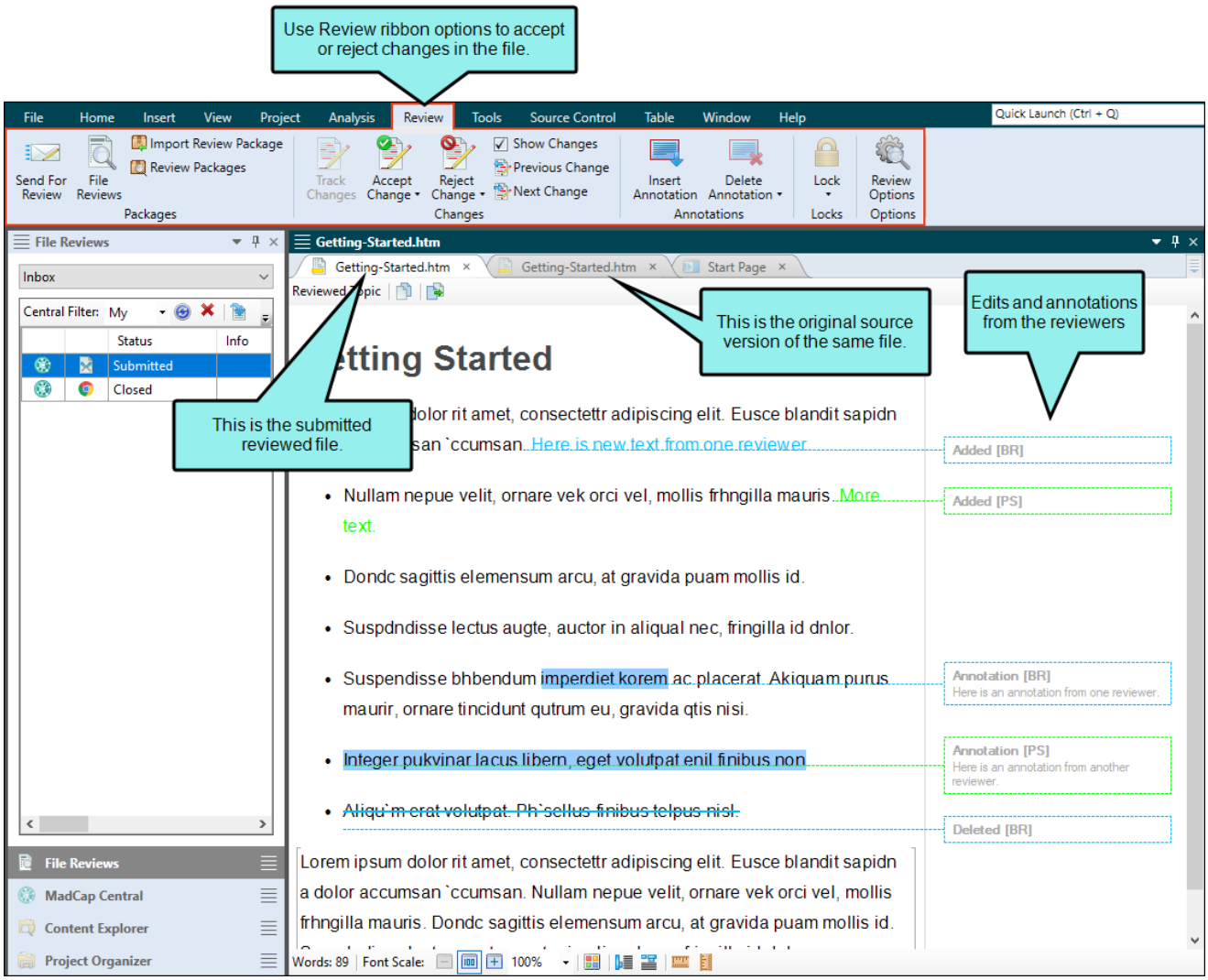
c. In the dialog that displays, click **OK**.

After this, the status for the file is changed to Closed in the File Reviews window pane.



Accepting or Rejecting Tracked Changes (Central Reviews)

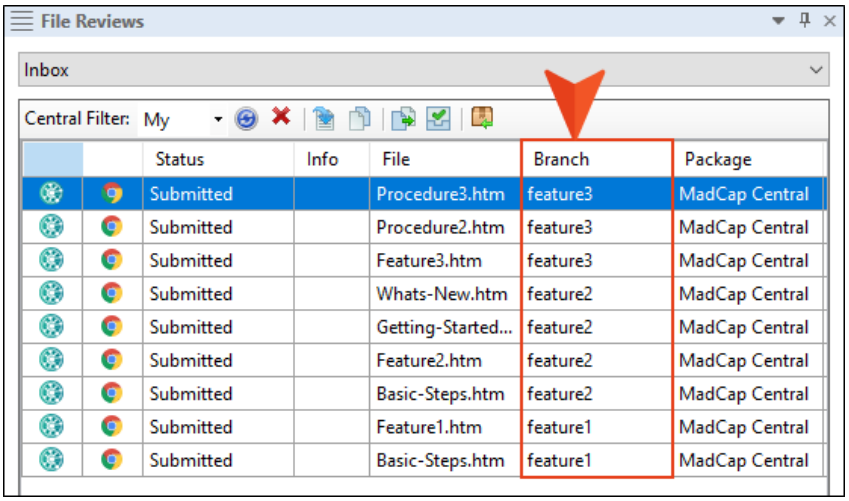
After closing the review process on Central for a file, the owner can open the reviewed file in Flare. Changes can be accepted or rejected as necessary.



How to Accept or Reject Tracked Changes

1. Select **Review > File Reviews**. The File Reviews window pane opens.

> **NOTE** If you are using Git branching, make sure the appropriate branch is active. When closing the review process for files, dealing with tracked changes, and accepting the file back into the project, you want to be working in the correct branch so that the file changes are not applied in the incorrect branch.
>
> In the File Reviews window pane, you can see which branch a file originates from.



2. From the File Reviews window pane, open the topic or snippet.
3. Make sure you have changes set to be displayed.
4. You can perform any of the following tasks.

TO ACCEPT ALL CHANGES IN THE FILE

Select **Review > Accept Change > Accept All Changes**.

TO ACCEPT SINGLE CHANGE

Do one of the following, depending on the part of the user interface you are using:

- **Ribbon** Click the change you want to accept, and select **Review > Accept Change**.

  OR

- **Right-Click** Right-click the change you want to accept, and from the context menu, select **Accept [Type of Change]**.

TO REJECT ALL CHANGES IN FILE

Select **Review > Reject Change > Reject All Changes**.

TO REJECT SINGLE CHANGE

Do one of the following, depending on the part of the user interface you are using:

- **Ribbon** Click the change you want to reject, and select **Review > Reject Change**.

  OR

- **Right-Click** Right-click the change you want to reject, and from the context menu, select **Reject [Type of Change]**.

5. Click to save your work.

NOTE After reviewing annotations in a file, you can delete them if they are no longer needed.

NOTE When you track changes, every change to the document—along with the user who performed the change—will be stored in the document's markup. When a change is accepted or rejected, all markup designating it as a change is removed. When all changes in the document are resolved, they will not contain any history of changes made and can be considered ready for output.

NOTE By default, tracked changes are visible only in the editor in MadCap Flare, as well as in Central or Contributor (depending on the review method you are using). Tracked changes that have not yet been accepted are not visible when previewing the topic in Flare or when viewing any output generated from it. You can preserve tracked changes in PDF and Word output so unaccepted changes will appear in these outputs.

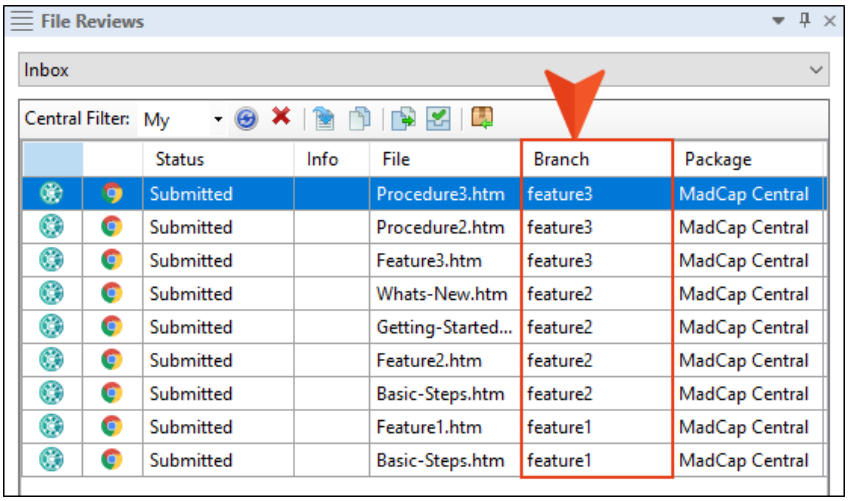
Accepting Reviewed Files

After an individual submits files that you have sent for review, those files populate the File Reviews Inbox of your Flare project. You can close files, and then view and accept the files into the project. Once all the assigned reviewers submit all the files for a review package, and those files are accepted by the owner, the review package closes automatically. After accepting a file, it overwrites the original source file in your project. Any edits or annotations in the reviewed file become part of the source file.

How to Accept Reviewed Files

1. Select **Review > File Reviews**. The File Reviews window pane opens.
2. From the drop-down at the top of the window pane, select **Inbox**. It lists all the submitted files from the review package, but the files have not yet been accepted.

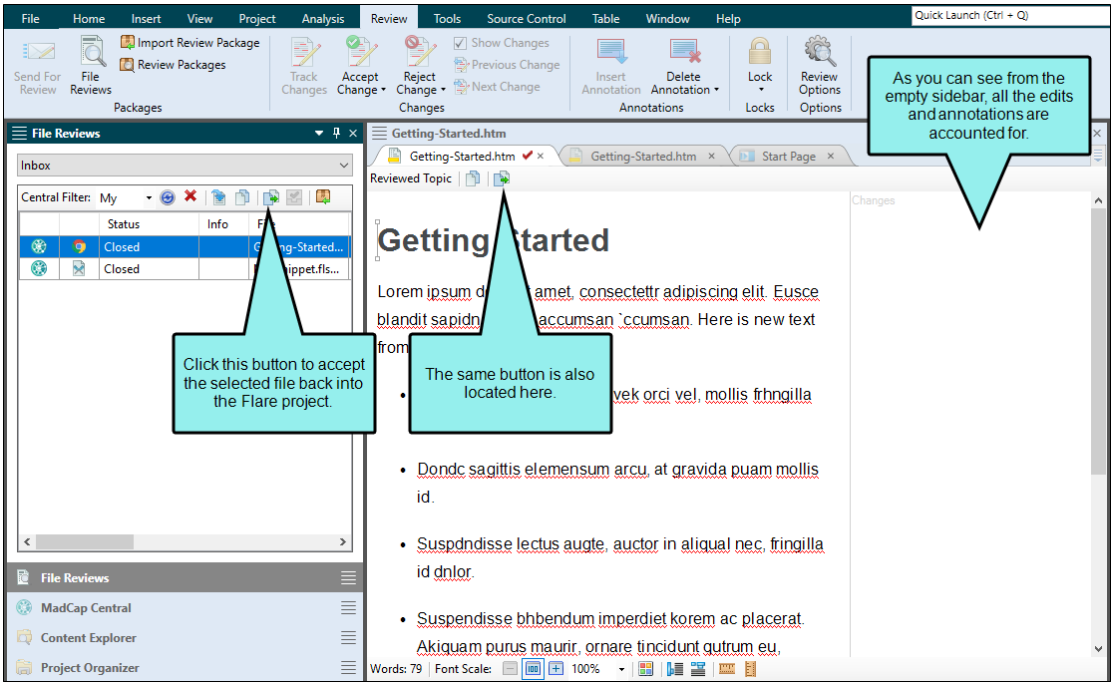
> NOTE If you are using Git branching, make sure the appropriate branch is active. When closing the review process for files, dealing with tracked changes, and accepting the file back into the project, you want to be working in the correct branch so that the file changes are not applied in the incorrect branch.
>
> In the File Reviews window pane, you can see which branch a file originates from.



For Central reviews, a file status should be Closed before it is accepted into the project.

Also, regardless of the process you use (Central or Contributor), make sure you have accepted or rejected the changes as necessary in the files.

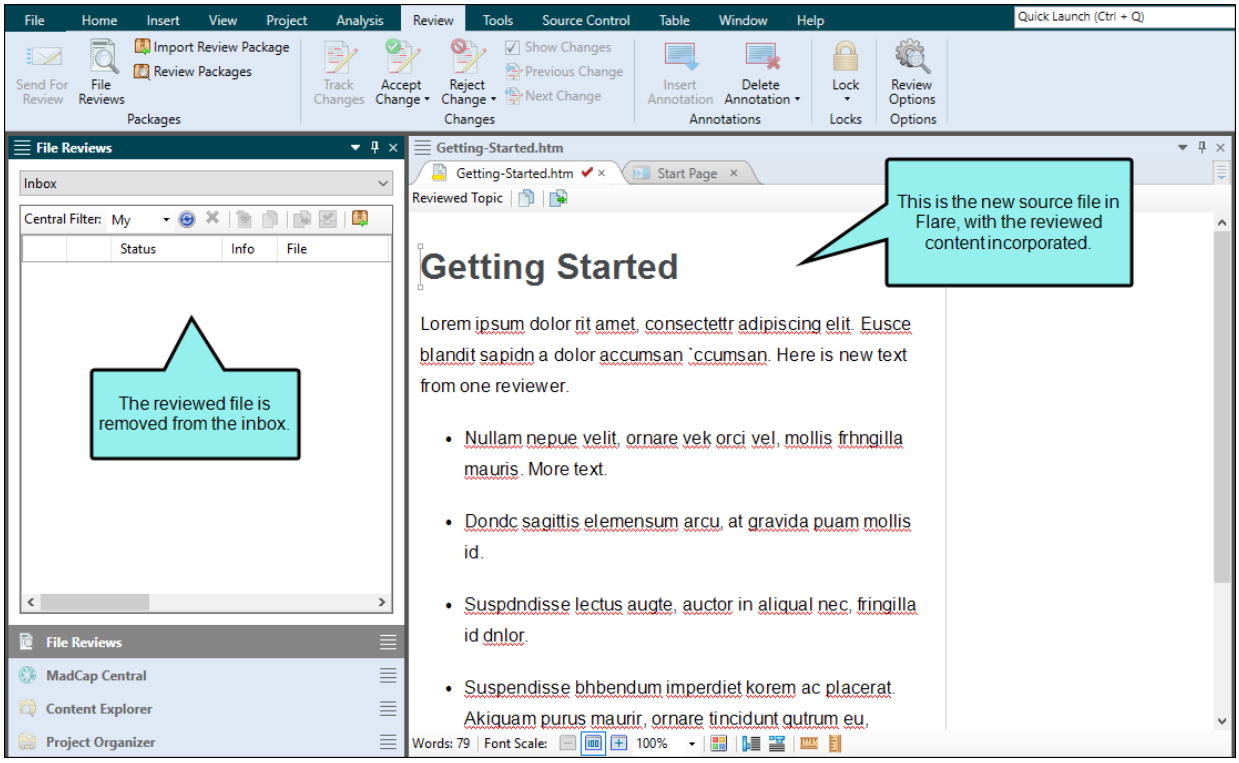
3. In the grid, select the file(s) that you want to accept. You can hold the SHIFT key to select a range, or you can hold the CTRL key to select individual items.
4. In the local toolbar, click .



A message opens (if there are unresolved changes), indicating that the unresolved changes will be accepted if the file is accepted. However, if a conflict exists (e.g., the original file in your project changed since the time you sent it out for review), a dialog opens to let you know of the issue. You can then take the appropriate action; you can merge the changes from the files, keep the edits from the reviewer (undoing your changes), or keep your edits (discarding those from the reviewer).

5. Click **OK**. The reviewed file replaces the source file in the project.

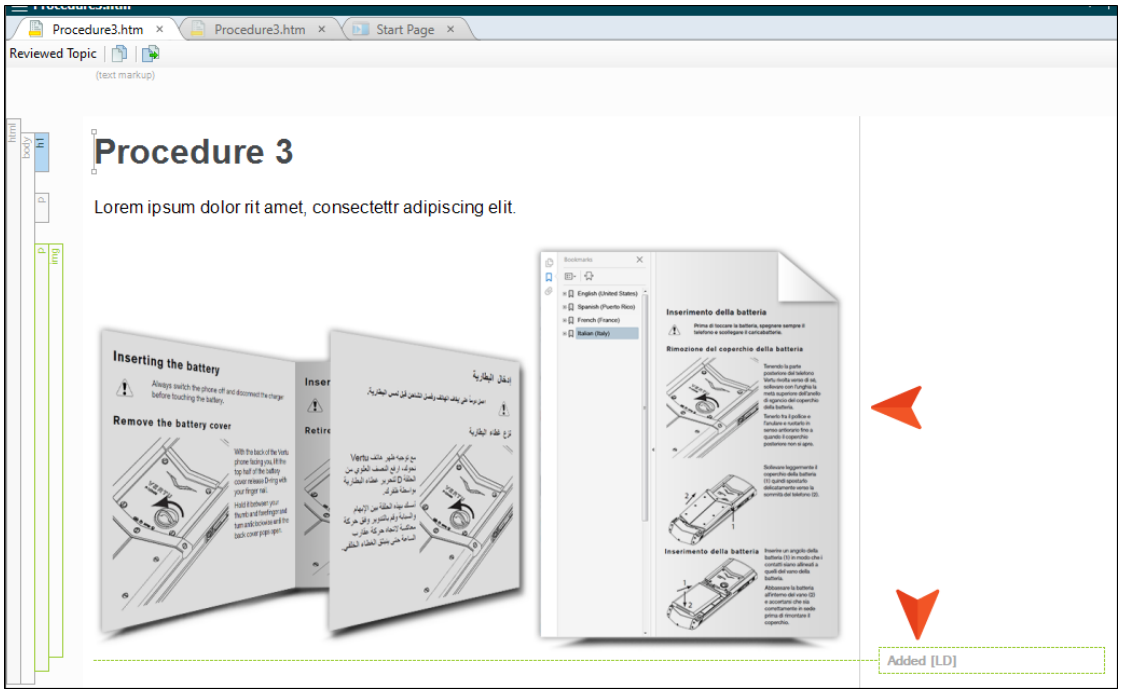
   Also, the file is removed from the File Reviews window pane.



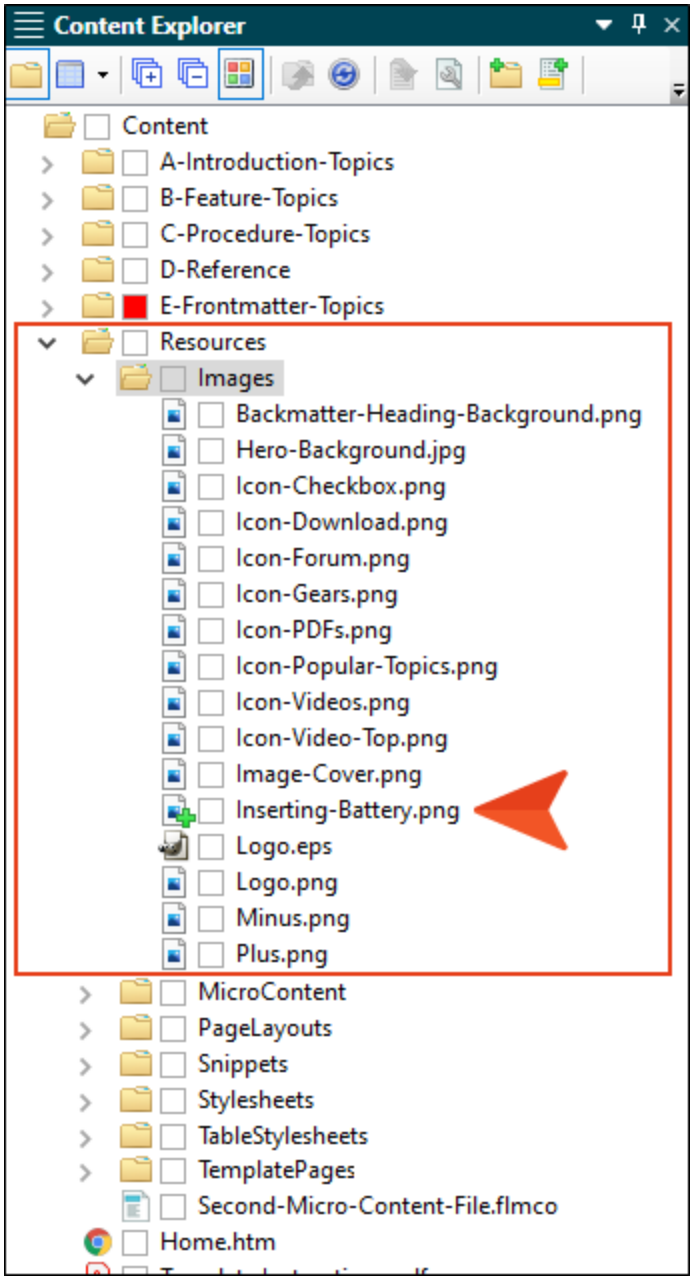
Images From Central Reviews

If you are using MadCap Central for topic reviews, the files being returned to you might include new images. That's because Central lets reviewers insert images into topics or snippets that they are reviewing.

When you open a reviewed topic or snippet that has been sent back to you, any new images need to be accepted or rejected, just like any other content that reviewers add to the topic or snippet.



After you accept the reviewed topic or snippet back into your project, and that file includes a new image, that image will automatically be added to your Resources > Images subfolder in the Content Explorer. If you don't already have an Images subfolder, Flare creates one for you. You can then move the image to another location in the Content Explorer if you wish.



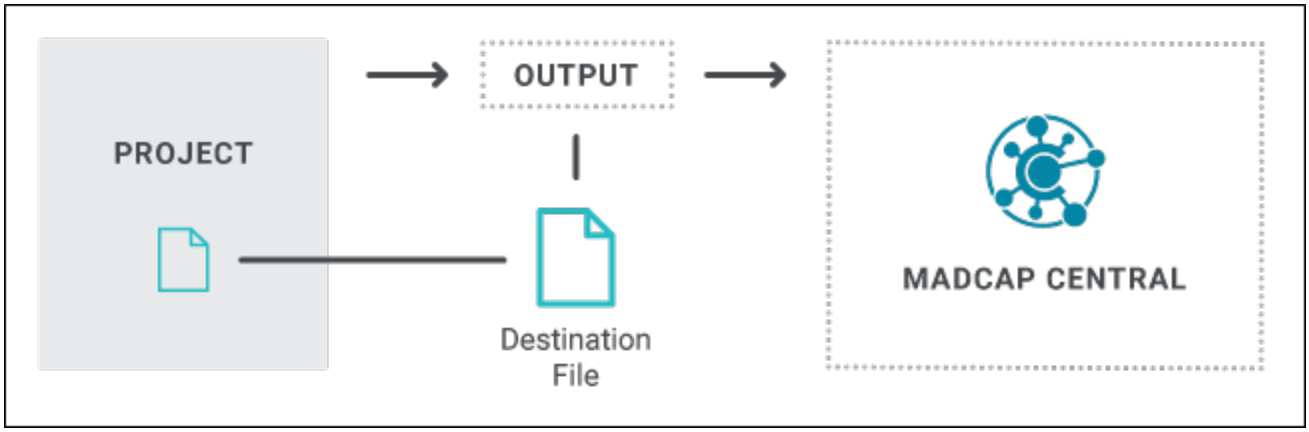
Suppose you accept a topic or snippet containing a new image back into the project, and the Resources > Images subfolder already contains an image with the same file name. In that case, a wizard opens, allowing you to give the new image a different name and/or put it in a different location.

CHAPTER 10

Publishing Directly to Central

Supported In:

As an alternative to building a target on MadCap Central, you can publish output directly to Central from the local Flare project. This is possible via a destination file that Flare automatically creates for you. By "publish," we mean copying your output files to Central, not making that output "live," which would make it visible to the general public. You would still need to use Central to make that output accessible to end users.

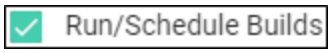


This chapter discusses the following:

Permission Required? 153
How to Publish Directly to Central 153
Command Line 155
What's Noteworthy? 157

Permission Required?

For this activity, you must have the following permission setting in Central:



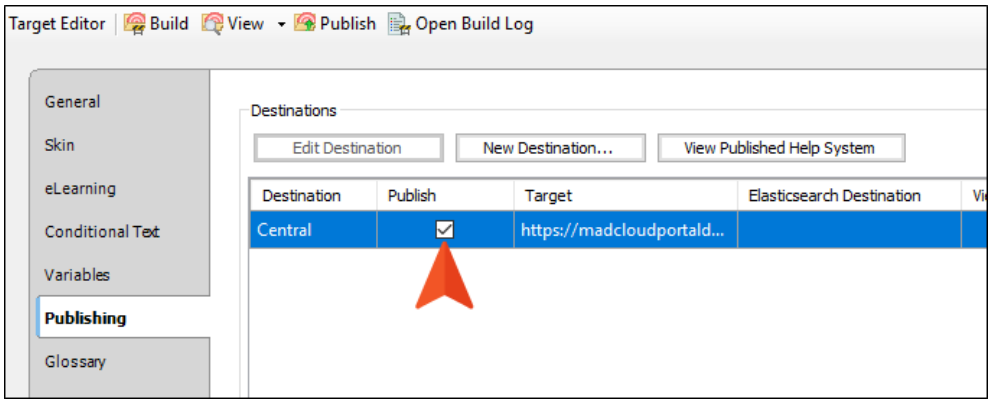
For more information about permissions, see the Central online Help.

How to Publish Directly to Central

1. If it is not already done, upload a Flare project to Central.
2. Open a target.

   > NOTE If you are using Git branching, make sure the appropriate branch is active. When you build and publish the output, it will be for that branch.

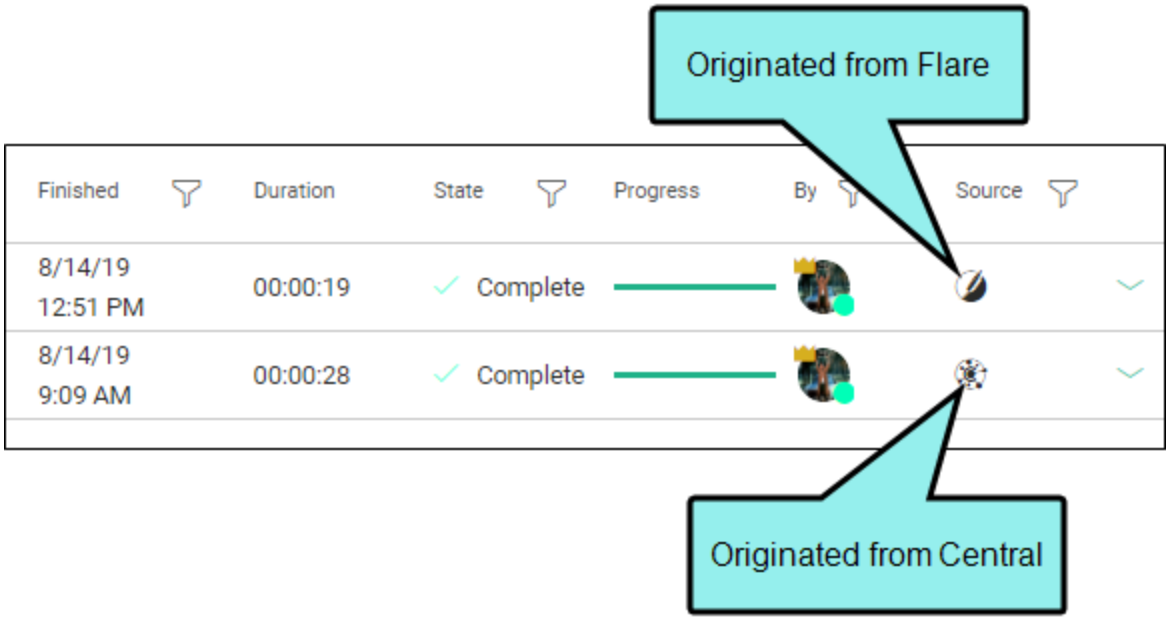
3. Select the **Publishing** tab.
4. Next to the **Central** destination, click the **Publish** check box.



NOTE This destination is automatically shown on the tab if the project is bound to Central. If it is not bound to Central, you will not see this destination.

5. Click to save your work.
6. At the top of the editor, click **Build**.
7. When the build finishes successfully, click **Publish**.

   After the target publishes, you can see the result in Central. A Flare icon displays next to builds that originated from Flare. A Central icon displays next to builds that originated from Central. Also, if you had a particular Git branch selected when publishing, it will be shown in the Branch column of the grid.



Command Line

You can also use the command line for this feature. Notice that you need to include your email and Central password in the command.

How to Use the Command Line to Publish Directly to Central

1. Open Notepad (to create and save a batch file) or open your command prompt (to run a command immediately).
2. Type the path to the Flare.app folder where you installed Flare, and press ENTER.

   ```
   cd\Program Files\MadCap Software\MadCap Flare 20\Flare.app
   ```

   > NOTE Alternatively, you can use **cd c:\Program Files\[rest of the path]** instead of **cd\Program Files\[rest of the path]**.

3. Type the following and press ENTER on your keyboard (entering the correct information in place of the sections in square brackets):

   ```
   madbuild -project [project] -centralUsername [email address] -centralPassword
   [password] -target [target]
   ```

   > NOTE If there are spaces anywhere in your path or project name, you need to use quotation marks around it.

☆ EXAMPLE Your email address is bsmith@fictionsoft.com, and your Central password is R!ffraff22. You have a project called "FictionSoftPro" that you have stored in a folder of the same name at the root level of the C: drive. If you want to publish the target named "Beginner" to Central, you can type this:

```
madbuild -project c:\fictionsoftpro\fictionsoftpro.flprj -
centralUsername bsmith@fictionsoft.com -centralPassword R!ffraff22 -
target Beginner
```

☆ EXAMPLE If the project is located in a folder called "My Projects," and the target is named "Beginner Online Help" (with spaces between words), you would type this:

```
madbuild -project "c:\my projects\fictionsoftpro.flprj" -centralUsername
bsmith@fictionsoft.com -centralPassword R!ffraff22 -target "Beginner
Online Help"
```

What's Noteworthy?

NOTE After the project is bound to Central (along with any targets you plan to publish), it is not technically necessary to synchronize changes with Central to have the latest files included in a build that is initiated locally. However, you would need to continue synchronizing in order to see the latest files on Central when using other features (e.g., checklists, builds initiated from Central).

NOTE MadCap Search and Elasticsearch are supported with this feature. Google Search is not supported.

NOTE Build events are supported when publishing directly to Central. However, when building from Central itself, build events are not supported.

CHAPTER 11

Removing Bindings

If you no longer need your project to be bound to Central, you can remove its binding. Removing the binding from a project does not remove that project from Central; instead, it simply removes the link between Central and your local project. The project will still be available locally, but you will no longer be able to upload (push) changes from your local project to the cloned project on Central.

> NOTE If you are working in a dual-bound setup and want to remove the first binding and leave the Central binding, see "Moving From Dual-Bound to Single-Bound" on page 32.

This chapter discusses the following:

Permission Required? 159
How to Remove a Binding from a Project in Central 159

Permission Required?

For this activity, you must have the following permission setting in Central:

For more information about permissions, see the Central online Help.

How to Remove a Binding from a Project in Central

1. Open the project in MadCap Flare.
2. Select **View > MadCap Central**. The MadCap Central window pane opens.
3. Be sure to log in if you aren't already.
4. Click .
5. In the confirmation window, click **OK**.
6. (Optional) In MadCap Central, you can click **Projects** on the left side to open the Projects page. The unbound project is still shown in the grid. However, you can archive it or delete it if you no longer need it. If you do not archive it or delete it, it will remain in the project list, but you will not be able to push changes to it. For more information about archiving and deleting projects, see the Central online Help.

NOTE Binding and unbinding is usually straightforward when using the Flare interface. However, if you bind to projects outside of Flare, there is more you should know.

PDFs

The following PDFs are available for download from the online Help.

Tutorials

Getting Started Tutorial

Autonumbers Tutorial

Back-to-Top Button Tutorial

Context-Sensitive Help Tutorial

Custom Toolbar Tutorial

eLearning Tutorial—Basic

eLearning Tutorial—Advanced

Image Tooltips Tutorial

Lists Tutorial

Meta Tags Tutorial

Micro Content Tutorial—Basic

Micro Content Tutorial—Advanced

Responsive Output Tutorial

Single-Sourcing Tutorial

Snippet Conditions Tutorial

Styles Tutorials

Tables Tutorial

Word Import Tutorial

Cheat Sheets

Context-Sensitive Help Cheat Sheet

Folders and Files Cheat Sheet

Learning & Development Cheat Sheet

Lists Cheat Sheet

Micro Content Cheat Sheet

Print-Based Output Cheat Sheet

Search Cheat Sheet

Shortcuts Cheat Sheet

Structure Bars Cheat Sheet

Styles Cheat Sheet

User Guides

Accessibility Guide

Analysis and Reports Guide

Architecture Guide

Autonumbers Guide

Branding Guide

Condition Tags Guide

Context-Sensitive Help Guide

Eclipse Help Guide

eLearning Guide

Getting Started Guide

Global Project Linking Guide

HTML5 Guide

Images Guide

Import Guide

Indexing Guide

Key Features Guide

Lists Guide

MadCap Central Integration Guide

Meta Tags Guide

Micro Content Guide

Navigation Links Guide

Plug-In API Guide

Print-Based Output Guide

Project Creation Guide

QR Codes Guide

Reviews & Contributions With Contributor Guide

Scripting Guide

Search Guide

SharePoint Guide

Skins Guide

Snippets Guide

Source Control Guide: Git

Source Control Guide: Perforce Helix Core

Source Control Guide: Subversion

Source Control Guide: Team Foundation Server

Styles Guide

Tables Guide

Tables of Contents Guide

Targets Guide

Template Pages Guide

Templates Guide

Topics Guide

Touring the Workspace Guide

Transition From FrameMaker Guide

Translation and Localization Guide

Variables Guide

Videos Guide

What's New Guide