NRCS

Natural Resources Conservation Service

Metis Tech Docs
Authoring Guide

July 31, 2024

1.0

USDA

Table of Contents

2 Document Purpose and Scope

Welcome to the *Metis Tech Docs Authoring Guide* (AG). The AG provides guidance on language and style conventions, as well as how-to procedures to support the Metis Tech Docs team.

The *Microsoft Style Guide* (MSG) and the *FPAC Design System* are the official style guides of Metis Tech Docs. The AG supplements these guides, providing guidance in cases where the aforementioned guides don't meet the needs of the Metis Tech Docs team.

The procedures in the AG describe using MadCap Flare, the authoring software Metis Tech Docs uses. The AG supplements, but does not replace the Madcap Flare help documentation.

The Metis Tech Docs team uses Bitbucket repositories for source control with Flare projects. The AG describes using Git Bash and MadCap Flare to perform tasks related to source control. Source control is a complex topic, and the AG only provides what a writer needs to know to complete a task. Writers may find it useful to read about Bitbucket and Git to understand the concepts behind the procedures in the AG.

Contributing to the Authoring Guide

The AG was created using MadCap Flare and uses a Bitbucket repository for source control. The AG is a living document and new content is frequently added. The best way to always have the latest version of the AG is to clone the Metis-TechDocs-Authoring-Guide repository and do frequent pulls.

Writers are encouraged to add content to the AG, but be sure to follow the guidance in the AG when contributing. In addition, you can contact Kate Erickson (kathryn. erickson) with feedback or suggestions. Just remember to be nice.

3 Language and Grammar

The following topics provide guidance on grammar and choosing the right words:

3.1 Addressing Our Users

Our users are people. This topic provides guidance for communicating with our users.

Address users in the second person. Use the pronouns *you* and *your*.

- *Enter your user name and password when prompted.*
- *The list updates as you enter text.*

Occasionally, you may need to specify who can use a feature, or you may need to provide conceptual information on how a certain type of user uses a feature. In these cases, it's okay to use a specific persona as the subject.

- *An Easement State Program Manager has unique privileges to approve or terminate an active CUA.*

Use the imperative mood when writing procedures. When possible, tell the user *what* to do instead of what they *can* do.

- *Use the Edit Vertices feature to move or delete one or more vertices of an unsaved boundary.*
- ~~*You can use the Edit Vertices feature to move or delete one or more*~~...

Keep your writing conversational, but avoid slang. Use contractions. It's okay to end a sentence in a preposition if it sounds more natural.

- *The Filter By State list contains all states you have access to.*
- ~~*The Filter By State list contains all states to which you have access.*~~

Write short, simple sentences. Break up lengthy or complex sentences. Omit unnecessary words, but not at the expense of sounding unnatural.

Use plain language whenever possible. Avoid technical terms or jargon that users might not be familiar with, including the names of UI elements.

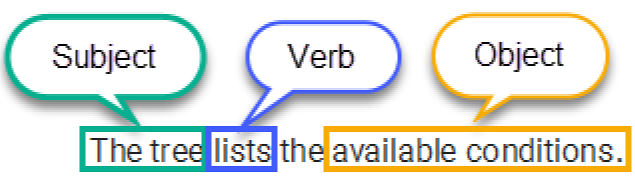
Write for an eighth-grade reading level. Although our users likely read at a much higher level, this is the industry standard.

> **Note**: MadCap Flare provides readability analysis. Select the **Tools** tab, and then select **Text Analysis**.

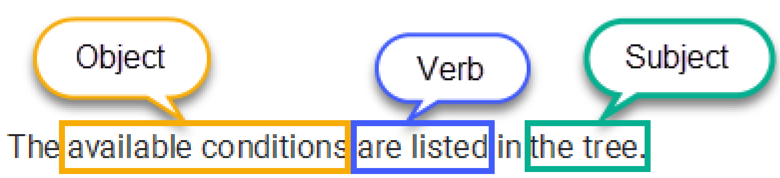
3.2 Active or Passive Voice

Voice is either active or passive. Keep it active whenever you can.

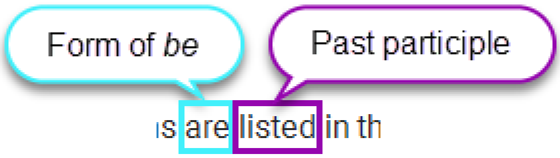
In an active voice sentence, the subject of the sentence performs the action.



In a passive voice sentence, the object moves to the subject position and receives the action of the verb.



Passive verbs typically consist of a form of the verb be plus the past participle form of a verb.



A form of "be" in a passive verb is one of the following words:

- Is, are
- Was, were
- Be (with modal auxiliary verb, such as, "The data may be overwritten")

A past participle verb is typically a verb plus the suffix *ed*:

- Interpreted
- Generated
- Formed

> **Note**: Some past participle verbs are irregular, for example, sat, known, frozen.

A preposition often follows the passive verb, and is a key clue to help you spot passive voice:

- Is generated *from*
- Was formed *by*
- Is copied *to*

If all else fails, insert *by zombies* after the verb. If the sentence still makes sense, you're using the passive voice:

- Ranking templates are used *by zombies* to create ranking pools.
- Filter on the date a question was last changed *by zombies*.

When to Use Passive Voice

Use passive voice to avoid an awkward or unnatural sentence that would be harder for readers to understand. This sometimes occurs when the subject is obvious or unimportant.

Examples:

Sites are evaluated for existing management and conservation efforts.

- *Set the Require Answer slider to Off if the question doesn't have to be answered.*
- *Set the Require Answer slider to Off ~~to not make users have to answer the question~~.*
- *Set the Require Answer slider to Off ~~if you don't want users to have to answer the question~~.*

Use passive voice to avoid personifying software or making the application sound omnipotent.

Examples:

- *Specific roles are permitted to edit.*
- *~~CART permits certain roles to edit.~~*
- *You may be prompted for your account login.*
- *~~CD may prompt you for your account login~~.*

Use passive voice to avoid condescending, scolding, or blaming the user.

Examples:

- *A Void Copy Mapping confirmation message appears if this change is attempted.*
- *~~A Void Copy Mapping confirmation message appears if you attempt this change, userboy.~~*

3.3 Capitalization

Metis Tech Docs uses title-case capitalization for headings, but sentence-style capitalization for everything else. With title-case capitalization, the first word of every word is capitalized. With sentence-case capitalization, everything is lowercase except the first word of a sentence and proper nouns.

Follow these capitalization guidelines:

- Use sentence-style capitalization most of the time. That means:
  - Capitalize the first word of a sentence.
  - Capitalize proper nouns. Proper nouns are the names of specific people, places, and certain things, such as titles of books.
  - Use lowercase for everything else.
- Always capitalize the first word of a new sentence. Rewrite sentences that start with a word that's always lowercase.
- Don't use all uppercase for emphasis. (It's OK to use italic sparingly for emphasis.)
- Only use internal capitalization if it's part of a name, such as eAuth.
- Don't capitalize the spelled-out form of an acronym unless it's a proper noun.

3.4 Punctuation

Punctuation is important because it affects comprehension, and incorrect punctuation can confuse readers. In technical documentation, most sentences should end with a period. Aim for short sentences that require minimal punctuation.

Refer to the Microsoft Style Guide (MSG) for guidance on punctuation. This topic provides additional guidance on the following:

Apostrophes

Colons

Comma

Hyphens

Punctuation to Use Sparingly

Punctuation to Avoid

Apostrophes

Use apostrophes in contractions and to make nouns possessive. However, avoid the following:

- Avoid awkward or ambiguous contractions, such as:
  - There'd
  - It’ll
  - It'd
  - They’d
- Avoid forming a contraction from a noun and a verb, for example, "CD's saving your changes."
- Avoid making proper nouns and acronyms possessive, for example, "CART's interface provides a search."

Colons

Use a colon at the end of a phrase that introduces a list. Otherwise, avoid using colons. If a heading or label in the UI has a colon at the end, omit the colon in instructions.

Comma

Use the serial, or Oxford, comma between items in a series. If a series contains many items, consider using a bullet list, especially if the items in the series are long. If a sentence doesn't contain a series and has more than one or two commas, the sentence may be too complex. The following sentence is an example of an overly complex sentence with multiple commas that should be broken up into shorter sentences:

"CD, for each of the features that you did not select (the other portions of the exploded multi-part practice), displays the status as Alternative and assigns it a new ID; CD auto-populates all other attributes for each of these practices from the original practice feature before the explode operation, except for Amount."

Hyphens

Hyphens are tricky. If you're not sure if you should use one, refer to the MSG for guidance.

Punctuation to Use Sparingly

Em Dashes

Em dashes (—) can function like a comma, a colon, or parenthesis. Use em dashes to write a sentence that would otherwise require two somewhat redundant sentences.

For example, instead of writing, "The three layer types are feature, reference, and basemap. The data for the three layer types comes from web services," you could write, "The data for the three layer types—feature, reference, and basemap—comes from web services."

However, bear in mind that it's often better to write two slightly redundant sentences than a long or complex sentence with em dashes.

En Dashes

En dashes indicate a range of numbers. Use an en dash if you're referencing text in the UI that uses an en dash, such as, "2020 – 2021." Otherwise, use the word "to."

Parentheses

Use parentheses to show an acronym following the written-out term. Otherwise, use parentheses sparingly, as they can make sentences too complex, especially when used in the middle of a sentence.

For example, it would be better to break up the following sentence into two sentences and omit the parentheses: "Depending on the document type configuration, you can overwrite a document (in other words, replace the latest version of a document with another version) by performing the following steps."

Don't use parentheses with a singular noun to indicate the plural form, as in, "Select the land unit(s)." In most cases, just use the plural form of the noun, as in, "Select the land units."

Quotation Marks

Don't use quotation marks to indicate text that appears in the UI. Instead, use the character style **span.UIText**.

Use italics, not quotation marks, to introduce a new term that you define right away.

Semicolons

Most of the time semicolons are unnecessary and can make a sentence too complex.

Punctuation to Avoid

Use words rather than symbols. Avoid the following types of punctuation:

- Ampersands
- Angle brackets (<>)

> **Note**: Don't use angle brackets as shorthand in instructions (Select **File** > **New**). Screen readers read angle brackets as "less than" and "greater than," which may be confusing for non-sighted users.

- Asterisks
- Backslashes
- Ellipses
- Exclamation points
- Pound signs

- Question marks
- Slashes (Use "or" instead)

Exceptions It's okay to use the punctuation above if:

- You are referencing characters that appear in code or in the UI.
- You are indicating characters that aren't allowed, for example, "The **Identifier** field cannot contain the following characters: \ / : * ? < > | " . , ' % [ ]."

3.5 Describing UI Elements

The *Microsoft Style Guide* (MSG) says to avoid talking about UI elements, and instead, describe what the customer needs to do. However, for clarity, sometimes it's necessary to include the type of UI element, for example, button, checkbox, dialog, etc.

This topic provides guidance on how to refer to the following frequently used UI elements:

Window

For information on styles to use for UI elements, see the topic Formatting UI Elements.

Actions Menu (Kabob)

The Actions menu in NRCS applications is a kabob element . When clicked, the Actions menu provides a list of actions the user can take.

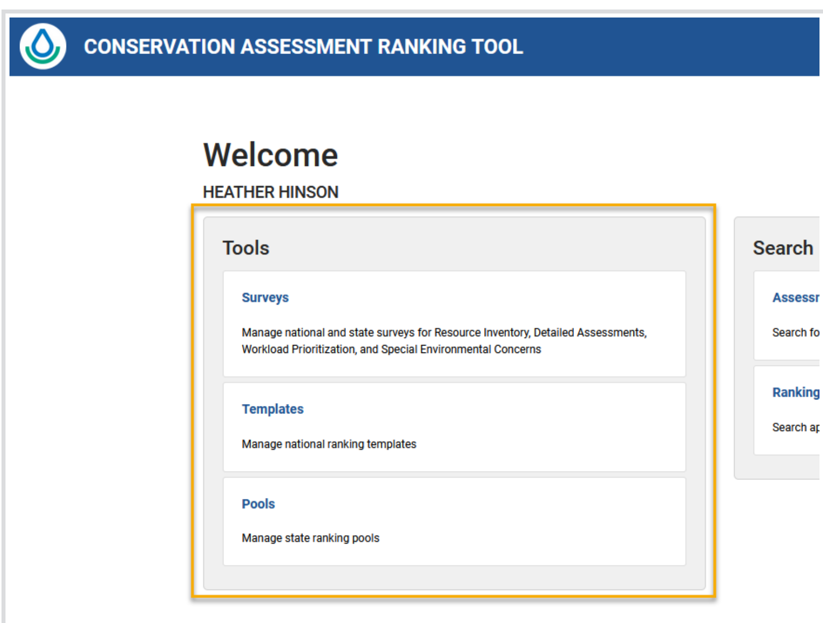
Use *Actions menu*. Don't use *kabob*, *vertical ellipsis*, or *three dots menu*.

Area, Section (Card)

A *card* is a UI element that groups related information in a container resembling a playing card.

Use *section* or *area*. Don't use *card*.

The following image shows the **Tools** card in CART.

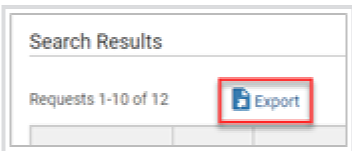


Button

When you click a button, something happens. Buttons can contain images (icon buttons) or text. Buttons can have a border, or have a border than only appears with hover.

☆ Use *button*. Avoid using *icon*, except for in the case of an icon button that users might not recognize as a button. If a button looks like a link, use *link*.

The following image is an example of a button that users might see as a link.



Checkbox

Checkboxes allow the user to select one or more items from a set.

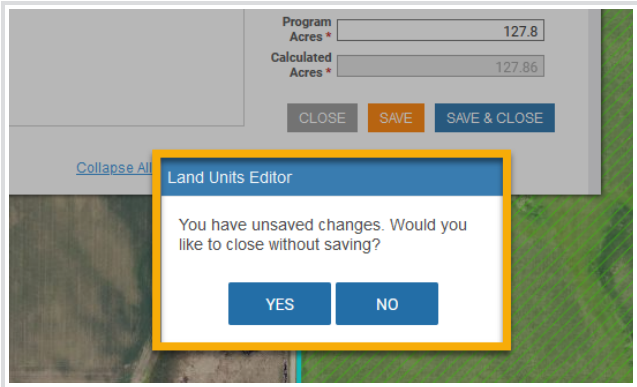
☆ Use *checkbox*. Use the verbs *select* and *clear*. Don't use *check box*. Don't use *deselect*, *mark*, *unmark*, *check*, or *uncheck*.

Dialog

A *dialog* is a fixed window that communicates information to the user and prompts them for a response. Modal dialogs require the user to respond before the user can do anything else.

☆ Use *dialog*. Don't use *dialog box*, *pop-up*, *popup*, *modal*, *modal dialog*, *screen*, or *window*.

The following image shows a modal dialog outlined in gold. Note that the dialog has the focus.



Dropdown Menu, Dropdown List

A dropdown menu lets a user choose one item from a list.

The MSG says to use *dropdown* as an adjective with a noun following it, as in *dropdown menu*, and to avoid using *dropdown* as a noun.

☆ Use *dropdown menu* or *dropdown list*. Don't use *drop-down*, *pull-down*, or *pulldown*.

Field

A field is a box where users enter text. For example, users enter their credentials in fields to log in to an application.

☆ Use *field*. Don't use *box*.

> **Note**: The MSG says to use *box* rather than *field*. However, Metis Tech Docs decided not to follow this guidance.

Image

☆ Use *image*; don't use *screenshot*, *illustration*, or *diagram*.

Option (Radio Button)

☆ Use *option*. Don't use *radio button*.

Page

☆ Use *page*. Don't use *screen*.

Panel

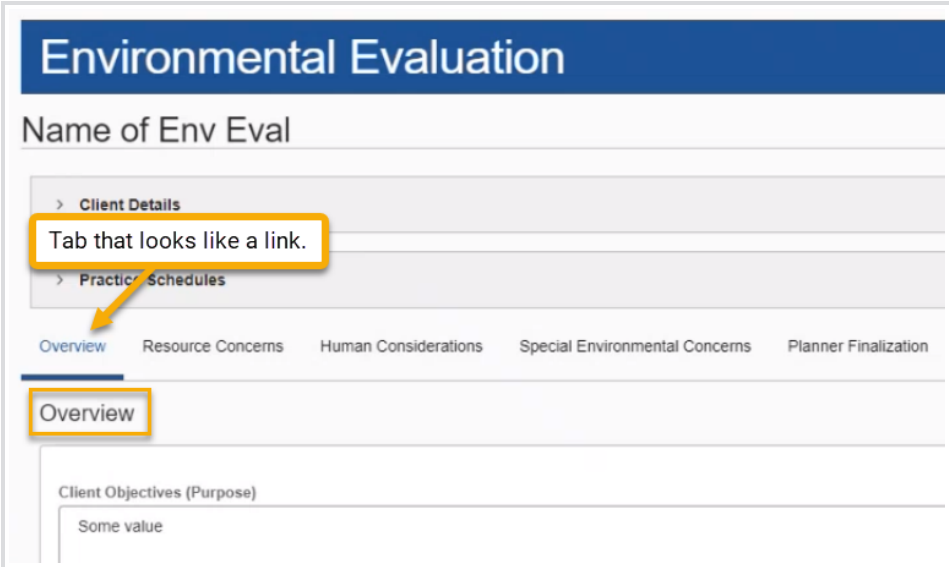
A panel slides out from the side, top, or bottom of the application window. The Table of Contents (TOC) in CD is a panel. In addition, a panel can have multiple sections.

☆ Use *panel*. Don't use *accordion*. For the CD TOC, use *section* for each of the items stacked in the panel.

Tab

☆ Use *tab*. Don't use *chevron* or *stepper*.

For tabs that don't look like tabs, use *page* in instructions. For example, the following image shows the Environmental Evaluation module in CD. When you hover over a tab, the text displayed in the tab looks like link text, so it might confuse users to call it a tab. In cases like this, omit the word *tab*, and simply refer to the **Overview** page, for example, "Select the **Overview** page and enter information in the **Client Objectives (Purpose)** field."



Table

There are many tables in NRCS applications. Some look like lists.

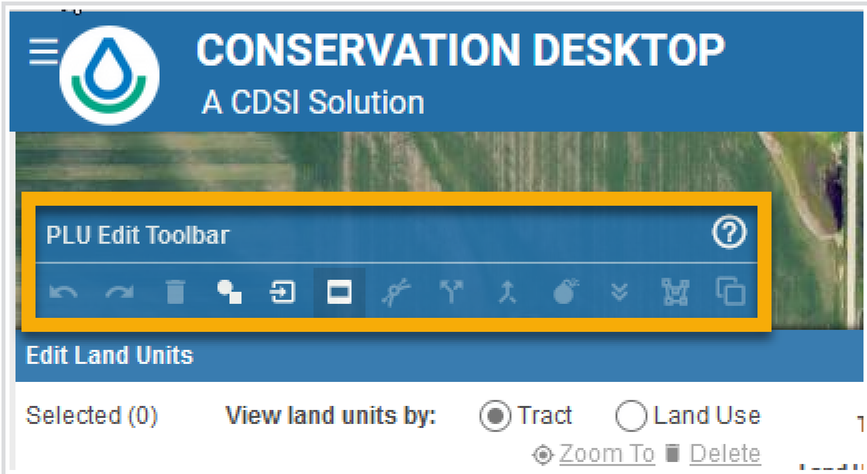
☆ Use *table*. Don't use *grid* or *table grid*.

Toolbar

A *toolbar* is a container for a set of related buttons. Toolbars usually display items in a single row or column.

☆ Use *toolbar*. Don't use *tool bar*.

In the following image, the **PLU Edit Toolbar** is an example of a toolbar in CD.

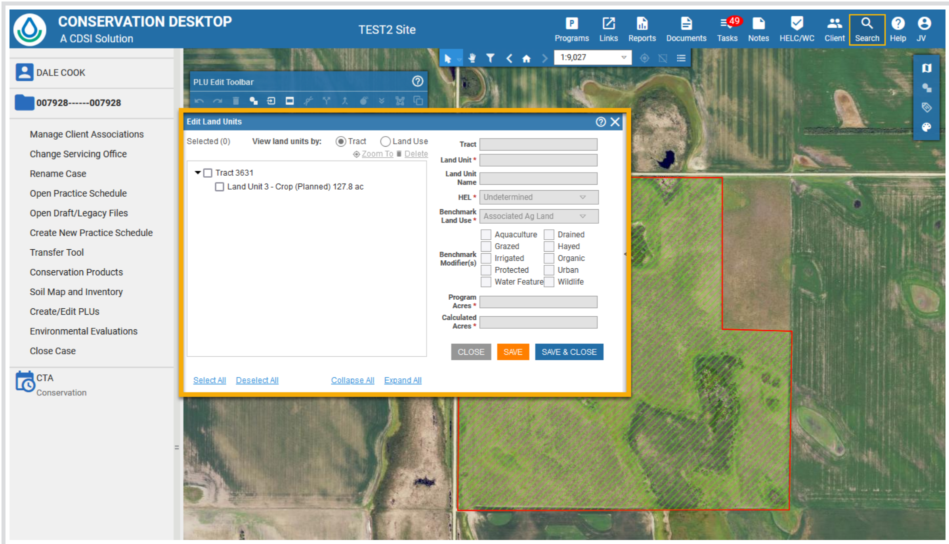


Window

A *window* is a container for displaying content separately from the main application interface. Windows are movable, often resizable, and don't prevent the user from interacting with the main application interface.

☆ Use *window* for containers that are are not [dialogs](). Also use *window* when referring to a browser window.

In the following image, the **Edit Land Units** window is an example of a window in CD. You can reposition the **Edit Land Units** window, and you can still interact with the side navigation, the map, or any of the toolbars.


CONSERVATION DESKTOP
A CDSI Solution
TEST2 Site
Programs
Links
Reports
Documents
Tasks
Notes
HELC/WC
Client
Search
Help
JV
DALE COOK
007928------007928
Manage Client Associations
Change Servicing Office
Rename Case
Open Practice Schedule
Open Draft/Legacy Files
Create New Practice Schedule
Transfer Tool
Conservation Products
Soil Map and Inventory
Create/Edit PLUs
Environmental Evaluations
Close Case
CTA
Conservation
PLU Edit Toolbar
Edit Land Units
Selected (0)
View land units by:
Tract
Land Use
Zoom To
Delete
Tract 3631
Land Unit 3 - Crop (Planned) 127.8 ac
Tract
Land Unit
Land Unit Name
HEL
Undetermined
Benchmark Land Use
Associated Ag Land
Benchmark Modifier(s)
Aquaculture
Drained
Grazed
Hayed
Irrigated
Organic
Protected
Urban
Water Feature
Wildlife
Program Acres
Calculated Acres
CLOSE
SAVE
SAVE & CLOSE
Select All
Deselect All
Collapse All
Expand All

4 Conventions

The following topics describe conventions that the Metis Tech Docs team has adopted:

4.1 File Names

Use the following guidelines for the file names of topics, snippets, images, and folders:

- Use title case.
- Separate words with a hyphen. Don't use spaces or other characters to separate words.
- Keep file names as short as possible while still making them meaningful.
- For long file names, omit prepositions (to, from, in, etc.), articles (a, an, the), and abbreviate words or leave out vowels.

File Name Examples

- File-Names.htm (this topic's file name)
- Frontmatter-Topics (folder name)
- Apply-Alt-Text.png (image name)
- Step-Apply-Style.flsnp (snippet file name)

Organizing Snippet Files

The Flare Help describes adding key words or letters at the beginning or end of certain names to make them easily identifiable and organized. For example, if you create snippets for individual steps that you will use many times, you could append each snippet with the word

"Step." For example, for the step "From the **Welcome** page, select **Surveys**," you could name the snippet file "Step-Welcm-Pg-Selct-Survys."

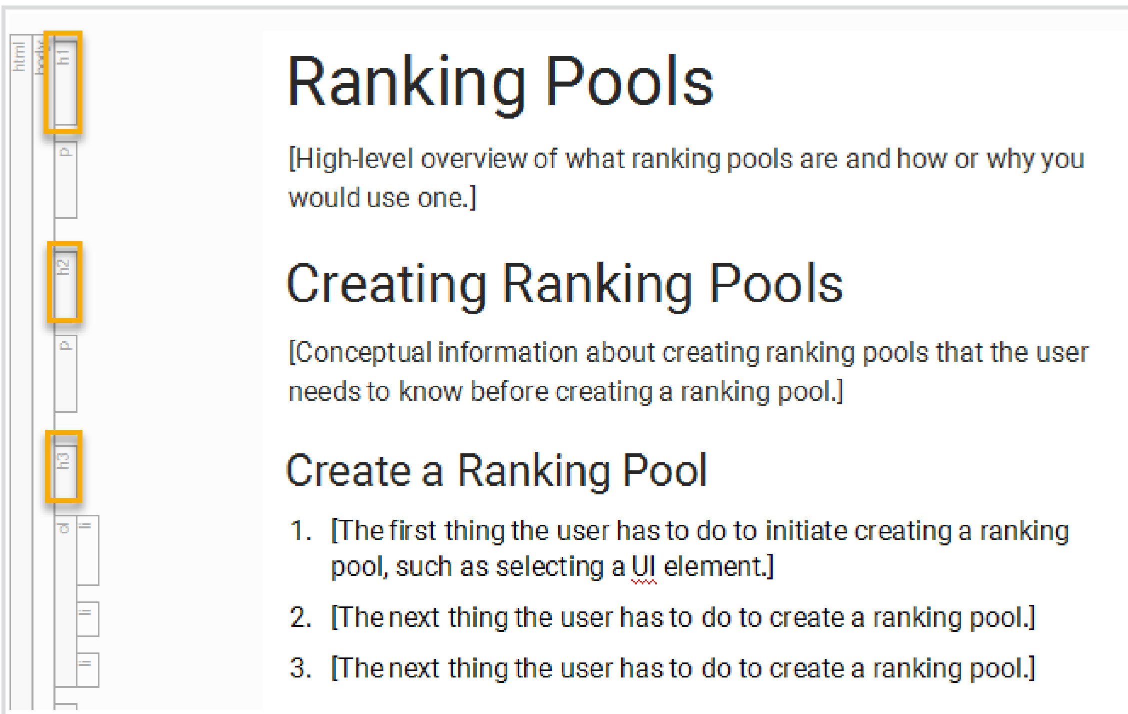
4.2 Headings

Headings are an important part of information architecture and reflect the hierarchy of topics. In addition, headings affect search results.

How you write a heading depends on the type of topic you're writing. In Metis Tech Docs' writing, all topics are one of the following kinds:

- Concept
- Procedure
- Reference

As you move from an h1 to an h2 to an h3, each new heading represents a new or more specific topic. Consider the following possible hierarchy of CART content:



In the above image, the h1 section provides an overview of ranking pools. This information applies to everything that comes next until get to another h1.

The h2 section is a conceptual topic that provides information specific to creating ranking pools. (A separate h2 section provides conceptual information about deleting ranking pools.) The user needs to know the conceptual information in this section before performing the procedure in the h3 section that follows.

The h3 section provides the procedure. Any conceptual information the user needs is in the previous h2 section.

Guidelines for All Headings

The following guidelines apply to all headings—conceptual, procedural, and reference:

- Use keywords in headings. This improves search results. Keep headings as short and specific as possible while still being meaningful.
- Use title case for headings; capitalize the first and last words, and capitalize all other major words (nouns, pronouns, verbs, adjectives, adverbs, and some conjunctions). The following items are additional capitalization rules for headings:
  - Lowercase the articles *the*, *a*, and *an*.
  - Lowercase prepositions, regardless of length, except when they are used adverbially or adjectivally (*up* in *Look Up*, *down* in *Turn Down*, *on* in The On Button, to in Come To, etc.).
  - Lowercase the common coordinating conjunctions and, but, for, or, and nor.
  - Lowercase to not only as a preposition, but also as part of an infinitive (to Run, to Hide, etc.), and lowercase as in any grammatical function.
- Use parallel sentence structure for headings at the same level whenever possible.
- Avoid having two headings in a row without text in between. However, don't insert filler text just to separate the headings.

Concept Topic Headings

For a conceptual heading, use one of the following constructions:

- Noun

  A word that represents a person, place, thing, or concept.

  Example: Accessibility

  > **Note**: A noun alone may not always be specific enough. Avoid using "Introduction," "Overview," or "About <topic>," as these headings aren't specific, and won't help with search results.

- Noun phrase

  A group of words, usually a noun in addition to a modifier, that functions like a noun.

  Example: Map-Related Tools

Procedure Topic Headings

For a procedure topic, use an imperative verb plus any other words that are necessary to show the scope of the procedure.

An imperative verb gives a command or is directive. The subject of an imperative verb is implied to be the second-person pronoun “you,” but the pronoun normally isn't included.

Examples:

- Edit Question Types
- Promote Text Changes to an Active Display Group

Reference Topic Headings

For a reference topic, use a noun or a noun phrase as described in Concept Topic Headings.

4.3 Character Styles

The topics in this section describe character styles to apply to text in certain situations:

4.3.1 Formatting UI Elements

Apply the character style **span.UIText** when referring to items in the UI, such as the labels of fields or buttons. The **span.UIText** style formats text with a bold font weight. This tells the reader that there is a UI element on the screen with the same label. For more information on the use of bold font, see Font Affects Reading.

Use the following guidelines for formatting text that appears in the UI:

- Use the style **span.UIText**.
- Match the case used in the UI.
- Avoid mentioning the type of UI element, such as "button" or "checkbox," unless it's necessary for clarity.
  - Select **Run Report**.
  - ~~Select the **Run Report** button.~~
- If a UI label includes a semicolon or ellipsis, omit the semicolon or ellipsis.
- When referring to a dialog box, use "dialog," not "pop-up window" or "dialog box." Use "dropdown" not "pulldown." See Describing UI Elements.

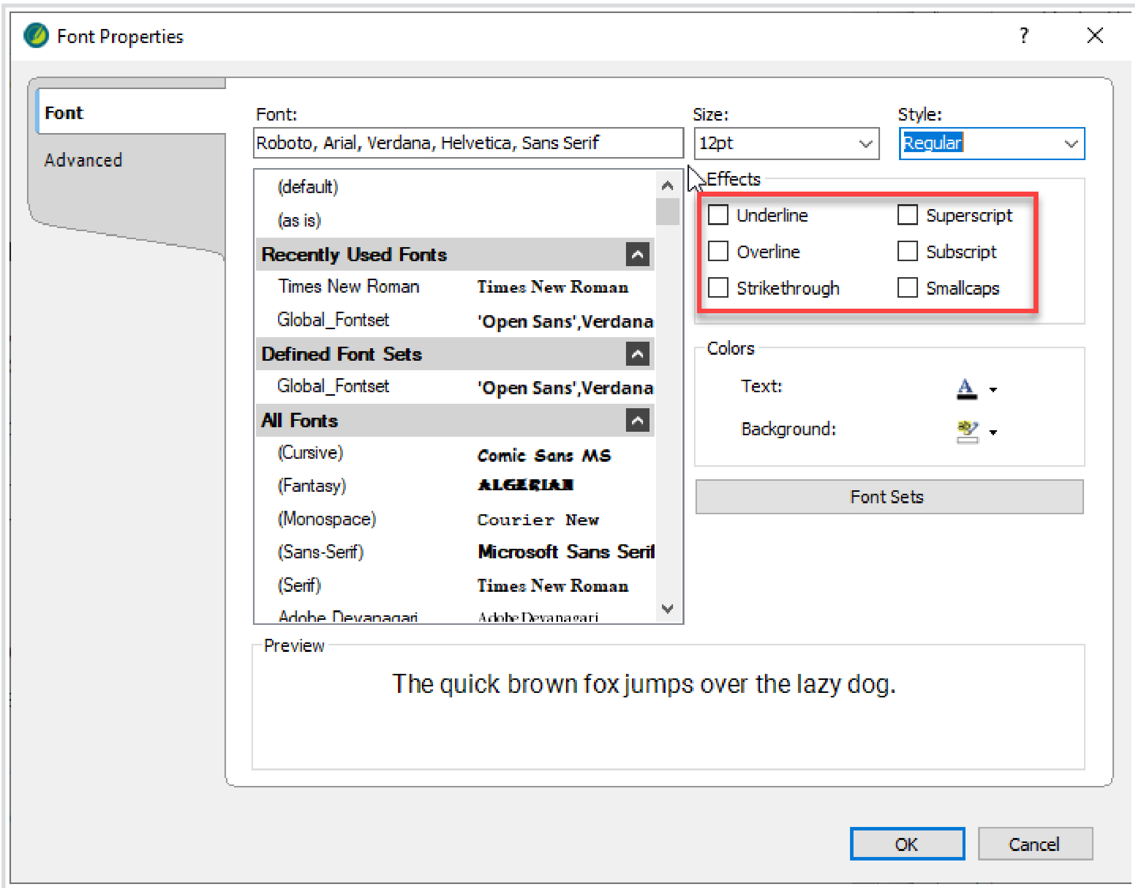
4.3.2 Emphasis and New Terms

Use the following guidelines for italic formatting:

- Use the style *span.Emphasis*.
- It's okay to use italic formatting sparingly for emphasis.
- Italicize the first mention of a new term if you're going to define it immediately in text. *Binding* a project means establishing source control for the project.

4.3.3 Other Special Text Formatting

In rare cases, you may need to underline text or apply strikethrough formatting. The **Font Properties** dialog (Ctrl+Shift+B) provides this type of text formatting.



4.3.4 Font Weight and Case Affect Reading

In technical writing, bold font weight is often used to signal to the user that an item is being displayed in the UI. Bold font catches the user's attention because it forces the eye to fixate on the letters in bold.

Consider the following sentences:

You want **to** be sure **that** your **users** can **read without** much **effort and** make **good** use **of** the **information** you want **to convey**.

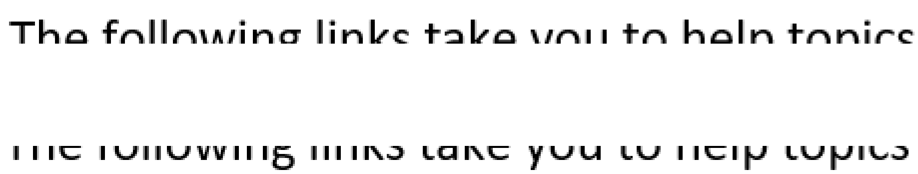
You **wa**nt **t**o **b**e **su**re **th**at **yo**ur **us**ers **ca**n **rea**d **wit**hout **mu**ch **eff**ort **a**nd **ma**ke **go**od **u**se **o**f **t**he **infor**mation **yo**u **wa**nt **t**o **con**vey.

In the first sentence, bold font alternates between regular font every one or two words.

The second sentence is written in Bionic Reading® style, a method that "facilitates the reading process by guiding the eyes through text with artificial fixation points." The words in bold font in the first sentence also act as artificial fixation points that alter the reading flow.

The point of these two sentences is to demonstrate that bold font truly does affect the reader's experience of the text. In limited quantities, bold font can help get the reader's attention, but if over-applied, the disruption it causes to the normal eye movements during reading can be bothersome to readers.

Along with font weight, case also affects the reading process. When we read, we look at the shape of words rather than each letter in each word. Research shows that the top half of a word helps us recognize the shape of the word more than the bottom half.



Changing the case of a word affects the shape of the word, particularly the top half that we use to recognize the word. This is why WORDS IN ALL CAPS or Words In Title Case are more difficult to read.

> **Note**: The MSG says that if you're not sure whether a word should be capitalized or not, don’t capitalize it.

With all of this in mind, pay attention to your use of bold font and case in your writing. Using case incorrectly, or inappropriately applying bold font makes text harder to read and understand.

5 Structuring Content

The topics in this section provide guidance on using the following elements to structure content:

5.1 Topic Types

The best practice is to organize information into concept, procedure, or reference topics. This approach is consistent with the Darwin Information Typing Architecture (DITA).

The topics in this section provide guidelines on how to structure the content you write.

5.1.1 Concept Topics

Use concept topics to:

- Describe a feature or capability.
- Provide an overview of a process.
- Provide background information.

Writing Good Concept Topics

Keep the following tips in mind when you're writing concept topics:

- Before you start writing, make sure you understand what the user needs to know or is trying to accomplish.

- Keep topics short and to the point. Write to a level of detail that is appropriate, and delete any unnecessary information.
- Aim for writing at the eighth-grade reading level.
- Use keywords consistently. It's okay to be repetitive with keywords.
- When appropriate, organize content into bullet lists to enhance scannability. See Bullet Lists for more information.

Concept Topic Headings

For the heading of a concept topic, use one of the following structures:

- Noun

  A word that represents a person, place, thing, or concept.

  Example: Accessibility

  > **Note**: A noun alone may not always be specific enough. Avoid using "Introduction," "Overview," or "About <topic>," as these headings aren't specific, and won't help with search results.

- Noun phrase

  A group of words, usually a noun in addition to a modifier, that functions like a noun.

  Example: Map-Related Tools

- Gerund phrase

  A phrase consisting of a gerund (verb root plus "ing") and any modifiers or objects associated with it.

  Example: Using the Ranking Pool Actions Menu

Create a Concept Topic

1. Press Ctrl+T on your keyboard.
2. In the **Add File** dialog:
    a. Under **File Type**, select **Topic**.
    b. In the **File Name** field, enter a name for the topic. Use a short, descriptive name in title case, and separate words with a hyphen (-). It's okay to abbreviate words in a long file name. See the topic File Names for more information.
    c. In the **1st Heading** field, enter the first heading for the topic. Often you can copy and paste the file name and then remove the hyphens. See the topics Headings and Topic File Names, Headings, and Titles for more information.
    d. Select **Add**.
3. Use short sentences to describe the conceptual information.
4. Save your work.

5.1.2 Procedure Topics

Procedural topics provide step-by-step instructions to help users complete a specific task.

Keep the following in mind when writing a procedure:

- Metis Tech Docs writes procedures for the "happy path" only. If there are multiple ways to perform a step, in general, just describe one way.
- Conceptual information that needs to accompany a procedure should precede the procedure in a separate section. In general, conceptual information should not interrupt a procedure.
- The shorter the procedure is, the better.

Procedure Topic Headings

For a procedural topic, use the imperative form of the verb plus any other words that are necessary to convey the scope of the procedure. Omit the infinitive "to."

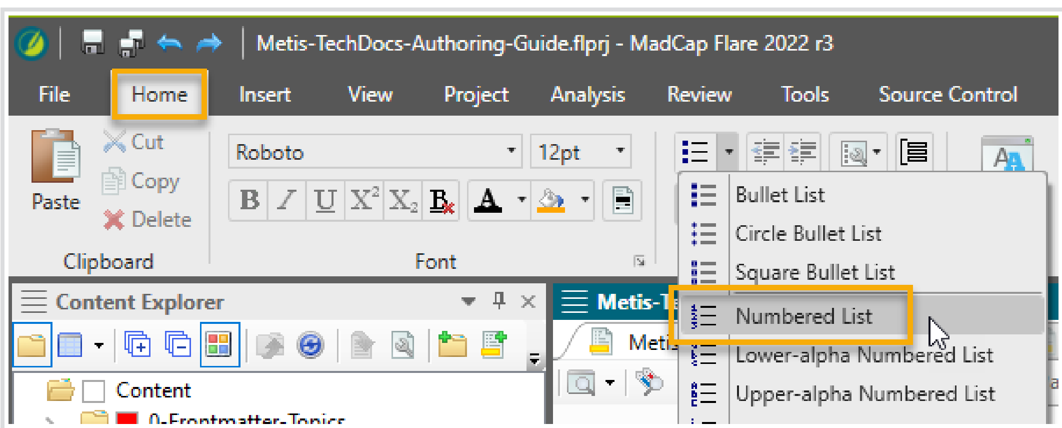
Examples:

- Edit Question Types
- Promote Text Changes to an Active Display Group

See the topic Headings for more information.

Create a Procedure Topic

1. Press Ctrl+T on your keyboard.
2. In the **Add File** dialog:
   a. Under **File Type**, select **Topic**.
   b. In the **File Name** field, enter a name for the topic. Use a short, descriptive name in title case, and separate words with a hyphen (-). It's okay to abbreviate words in a long file name. See the topic File Names for more information.
   c. In the **1st Heading** field, enter the first heading for the topic. Often you can copy and paste the file name and then remove the hyphens. See the topics Headings and Topic File Names, Headings, and Titles for more information.
   d. Select **Add**.
3. At the beginning of the new topic, add a *brief* description or provide a use case for the task if the preceding section doesn't provide this information.
4. Select the **Numbered List** option in the **Paragraph** area of the **Home** ribbon.



5. Enter text for the step, and then press Enter to move to the next step in the procedure.

6. (Optional) To move to the second level of the list (lower alpha), press Tab.

   An indented sub-step **a** is added.

7. To add another sub-step, press Enter.

   To move up a level, press Backspace or select the outdent button in the **Paragraph** area of the **Home** ribbon.

8. (Optional) To move to the third level of the list (lower roman), press Tab.

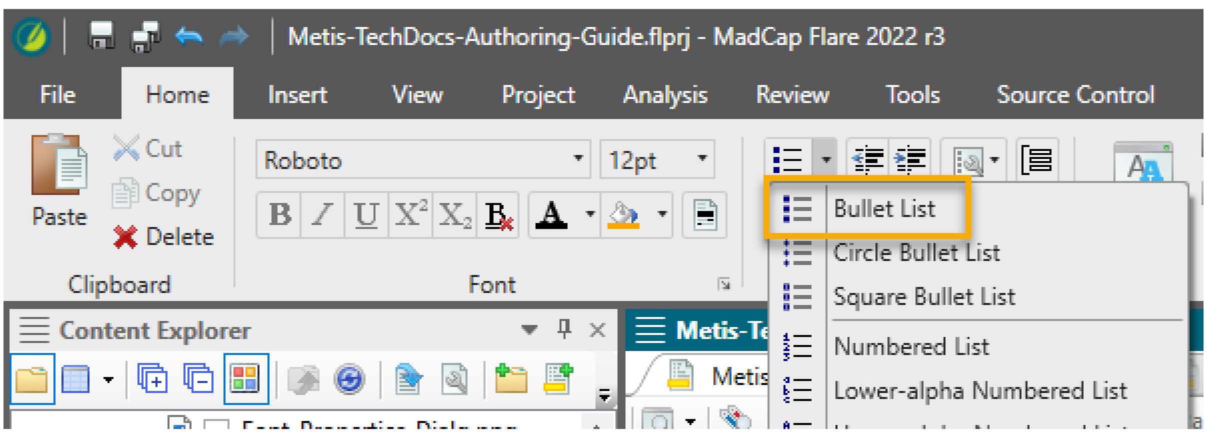
   A lower roman **i** is added.

   > **Note**: Flare won't let you add a third level with the Tab key if you haven't entered text at the second level.

9. Continue adding list items to complete the steps in the procedure.

10. (Optional) To change a sub-step to a bullet list item, select the generic **Bullet List** option from the list menu on the **Home** ribbon. Press Enter to add additional bullet list items.

    To return to the numbered list, press the outdent button on the Home ribbon.



11. If things get messy with numbering, and you can't get back to the highest level of the list, you can cheat a little as follows:

    a. On a paragraph line immediately following the list, select the **Numbered List** option on the **Home** ribbon. Flare adds a new <ol> with a step **1**.

    b. Right-click the <ol> tag bar that corresponds to the new list, and select **Continue Sequence**.

       Step **1** updates to the appropriate number where the previous numbered list left off.

12. Continue adding list items to complete the steps in the procedure.
13. Save your work.

For more information on working with lists, see the related topics:

- Lists
- Drag List Items to Reorder Steps
- Inserting an Image into an Ordered List

5.1.3 Reference Topics

Reference topics provide information that users tend to look up rather than memorize. Reference information often appears in lists or tables.

The following reference table from the CD Help describes user roles and their source.

Conservation Desktop Role	Role Source
Planner	Granted in zRoles
Planning Support	Granted in zRoles
Program Support Assistant	Granted in zRoles
Regional Conservationist	Granted in zRoles

Reference Topic Headings

For the heading of a reference topic, use one of the following structures:

- Noun

  A word that represents a person, place, thing, or concept.

  Example: Accessibility

  > **Note**: A noun alone may not always be specific enough. Avoid using "Introduction," "Overview," or "About <topic>," as these headings aren't specific, and won't help with search results.

- Noun phrase

  A group of words, usually a noun in addition to a modifier, that functions like a noun.

  Example: Map-Related Tools

5.2 Admonitions

Admonitions provide readers with information that isn't part of the flow of the text.

The thoughtful use of admonitions can add value to documentation. However, similar to "banner blindness," there's evidence that readers skip over clutter on a page, such as numerous admonitions.

Metis Tech Docs currently uses *note*, *important*, and *warning* admonitions, which appear as follows:

> **Note**: Provides information that is useful but not critical to the user.

> **Important!** Provides information that the reader should not overlook.

> **Warning!** Alerts the user that the current operation will cause data loss or irreversible changes to the current setup.

5.2.1 Guidelines for Admonitions

The following items are general guidelines for using admonitions:

- Use admonitions for information that adds value, but isn't part of the flow of what you're writing.
- Use admonitions sparingly.
- Place admonitions before the steps they apply to.
- Keep the information as brief and to the point as possible.
- Avoid "admonition sandwiches," or multiple admonitions in a row. If you absolutely must, put the most serious admonition first.

5.2.2 Note Admonitions

Use a note admonition to bring additional information to the user's attention. However, use note admonitions judiciously. If you're not sure whether something should be a note, write it first in regular text and then decide if it's truly noteworthy.

Use a note in the following cases:

- The information you're sharing is relevant and useful, but not necessary for what the reader is doing. If the reader skips the note, it won't cause problems.
- Interrupting the reader at this point is not an obstacle to the reader. For example, your note isn't suggesting an alternative that leads the reader down a different path.
- The information is not part of the flow of what you're writing—it's not just a continuation, a result, or a pointer to additional information.

Avoid the following with notes:

- Making a full procedural step into a note
- Using notes for cross-references
- Using multiple consecutive note admonitions

5.2.3 Important Admonitions

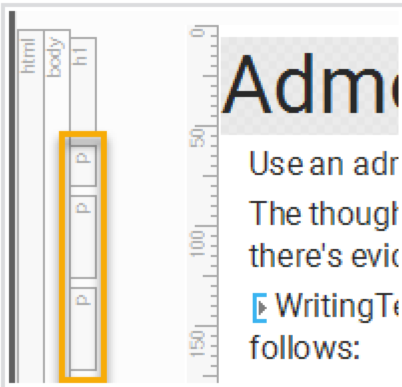
Use an important admonition for information that readers should not overlook. While this information might not alter what the user is doing, it should show the user that this piece of information could be vital.

5.2.4 Warning Admonitions

Use a warning admonition to alert readers that what they're about to do will alter their current setup. For example, use a warning before a step that causes database changes that can't be undone. A warning admonition should inform readers not to continue unless they are fully aware of the consequences.

5.2.5 Add an Admonition

1. Select a <p> tag bar, or place your cursor in the text of a <p> tag.



2. In the **Styles** window, select one of the following styles:
   - **p.Note**
   - **p.Important**
   - **p.Warning**

   The <p> tag displays with the admonition style you selected.

To insert an admonition into an ordered list, follow the steps in the topic Inserting an Image into an Ordered List.

5.3 Images

The topics in this section provide guidance on when and how to use images.

The judicious use of images can add value to documentation. When a procedure references a UI element, an image can help users find the UI element more quickly. This can improve the user experience by reducing frustration. However, if the UI is updated and the image isn't, the image can be unhelpful or even confusing.

This section provides the following topics:

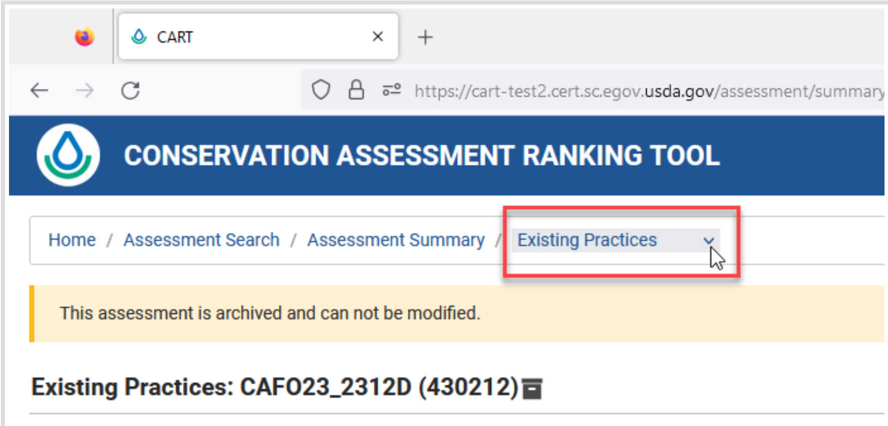
5.3.1 When to Use a Screenshot

Use a screenshot *only when it adds value*. Remember that screenshots compliment the text and never replace it. If you add a screenshot, make sure that readers can get the whole story from either the textual information or the image.

A screen shot may be appropriate in the following situations:

- In a pivotal or potentially confusing step, a screenshot may increase the user's chances of success.
- During a long or complicated procedure, one or more screenshots may help orient or reassure the user that they are where they should be.
- For unlabeled areas of the UI, a screenshot may help users find what they need more quickly. Remember that you still have to describe the location in the text.
- When the UI doesn't follow its usual pattern, a screenshot may be useful for guiding the user.

  The following image shows the only breadcrumb link in CART that has a dropdown menu. Using a screenshot in this case may help users, since this behavior is an oddity that might be harder to quickly understand from the textual description alone.



5.3.2 Screenshot Best Practices

The following list describes best practices for screenshots:

- Crop screenshots whenever possible. You don't usually need to include the entire screen, and cropping saves the user time figuring out what they're supposed to see. However, include enough of the screenshot to make it useful. If there is a lot on the screen, providing just a button might not be as helpful.
- Use screenshots that help users perform a step rather than screenshots that show what they should see if they performed the step correctly.
- Avoid screenshots of simple UI elements, such as progress bars, confirmation boxes, or message text. If it is important to include the text of a warning or error message, add it as regular text. The typed message is more searchable and 508-compliant.
- Avoid screenshots of UI dialogs or modals that have a help button. This is especially true when we launch our context sensitive help.

5.3.3 Inserting an Image

Metis Tech Docs uses images as follows:

- Inline icons that appear "in-line" with text, typically in procedures.
- Screenshots that illustrate conceptual or reference information, or that help users follow along in a procedure.

This topic provides the following procedures:

Insert an Inline Icon

Insert an Image

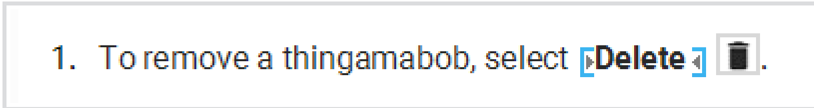
> **Important!** Insert an Image describes how to insert an image into a *paragraph*, which is different than inserting an image into a *numbered procedure*. For help inserting an image into a numbered procedure, see Images in an Ordered List.

Insert an Inline Icon

Use the following steps to

1. In the ribbon, select the **Insert** menu, then select **Image**.
2. In the **Insert Image** dialog:
   a. Select the image you want to insert.
   b. Provide a brief description of the image in the **Screen Tip** and **Alternate Text** fields.
   c. Select **OK**.
3. Select the image, then select the **Styles** panel:
   - For icons that are roughly square, apply the style **img.Toolbar_btn**.
   - For icons that are taller than they are wide, apply the style **img.Toolbar_btn-T**.
   - For icons that are wider than they are tall, apply the style **img.Toolbar_btn**.

   > **Note**: Doing a build and viewing the icon in the output is the best way to verify the optimal style for the icon.
4. Immediately in front of the icon image:
   a. Enter the textual name of the icon.
   b. Highlight the textual name and apply the **span.UIText** character style.



Insert an Image

> **Note**: For help inserting an image into a numbered procedure, see [Images in an Ordered List]().

1. In the XML Editor, place the cursor on an empty paragraph (<p>) line where you want to insert the image.
2. In the ribbon, select the **Insert** menu, then select **Image**.
3. In the **Insert Image** dialog:
   a. Select the image you want to insert.
   b. Provide a brief description of the image in the **Screen Tip** and **Alternate Text** fields.
   c. Select the ellipse next to **Style ID**, and then select **ImgBorder-90-percent**.

      > **Note**: Occasionally, an image may appear too large in the output. If this occurs, you may need to apply a different border image style, such as img.ImgBorder-35-percent.

   d. Select **OK**.

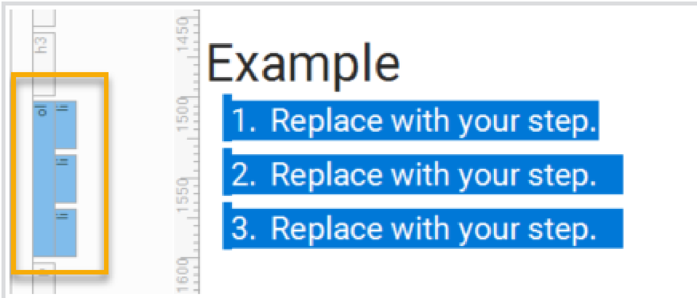
5.3.4 Inserting an Image into an Ordered List

This topic provides the steps for inserting an image into an ordered list on a new line, as opposed to inserting an inline image.

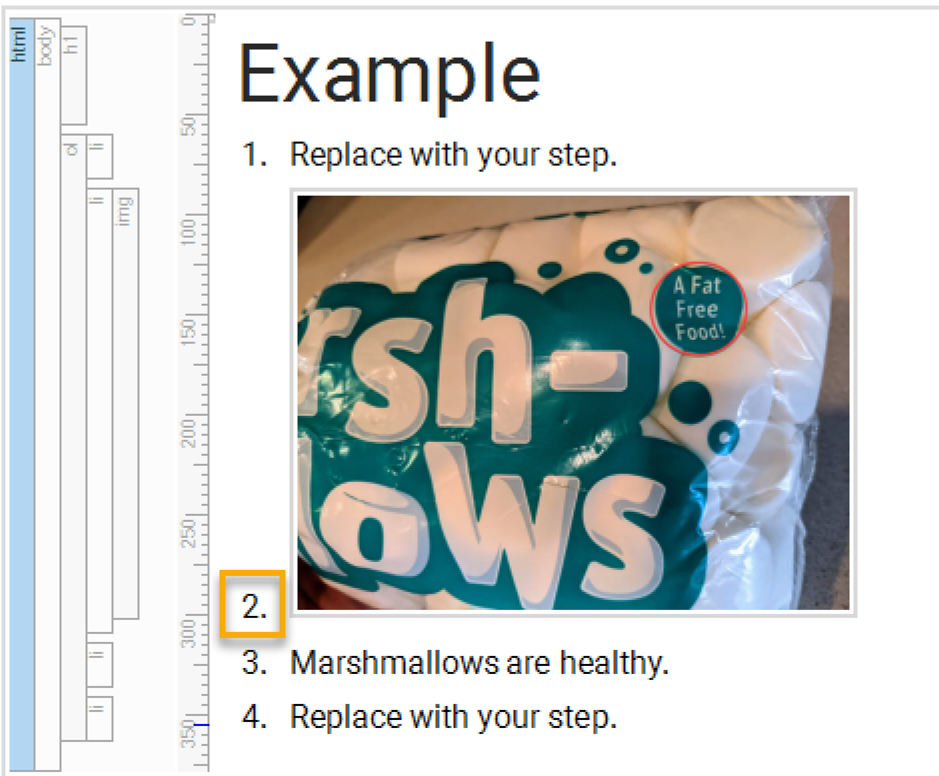
> **Note**: The method you use to insert an image into an ordered list is also how you insert an admonition or a table into an ordered list. So pay attention.

If you insert an image into an ordered list, the formatting won't be right unless you follow the steps in this topic.

The numbered steps in a procedure consist of an ordered list (<ol>) tag with list item (<li>) tags for each step.



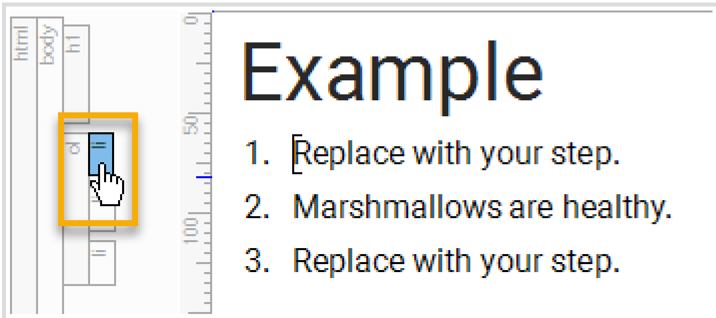
If you insert an image into an ordered list, the image will look like numbered step.



You can't format an image correctly unless the image is inside a <p> tag or a div. The following section provides the steps for inserting an image using the <p> tag method.

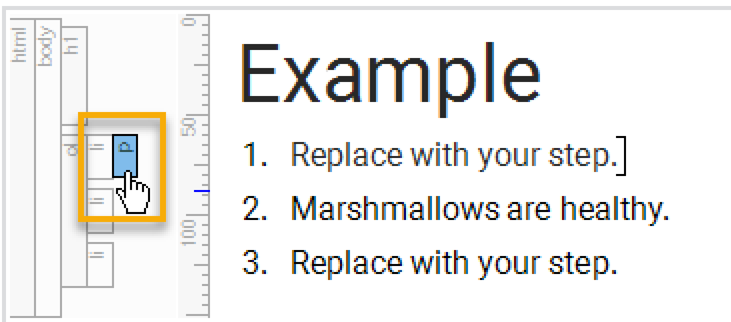
Insert an Image into an Ordered List

1. Locate the <li> tag bar that corresponds to the step where you want to insert the image.

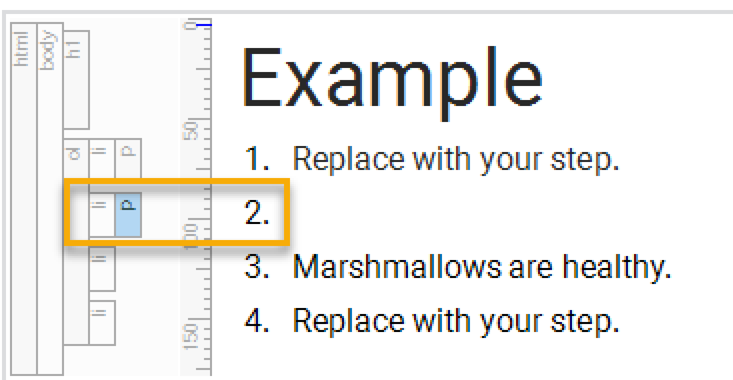


2. Right-click on the <li> tag bar, and select **Make Paragraph Item(s)**.

   A <p> tag is added next to the <li> tag.



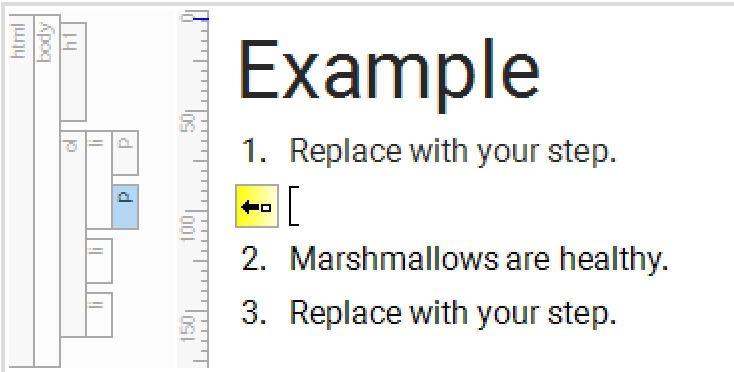
3. Place your cursor at the end of the step with the new <p> tag and press the <Enter> key.
   A new <li> tag with an embedded <p> tag is added, as well as a step number.



> **Note**: To remove an embedded <p> tag, right-click the <li> tag next to the <p> tag, and select **Make Simple Item(s)**.

4. Press the <Backspace> key once to remove the numbered step.

   A mysterious yellow square containing a black arrow displays.

5. Insert the image:

   a. With your cursor at the beginning of the line with the yellow box, press <**Ctrl+G**> or paste the image.

      The **Insert Image** dialog opens.

   b. Select the image you want to insert.

      > **Important!** Ensure that the image's file name follows the guidelines in File Names (on page 23).

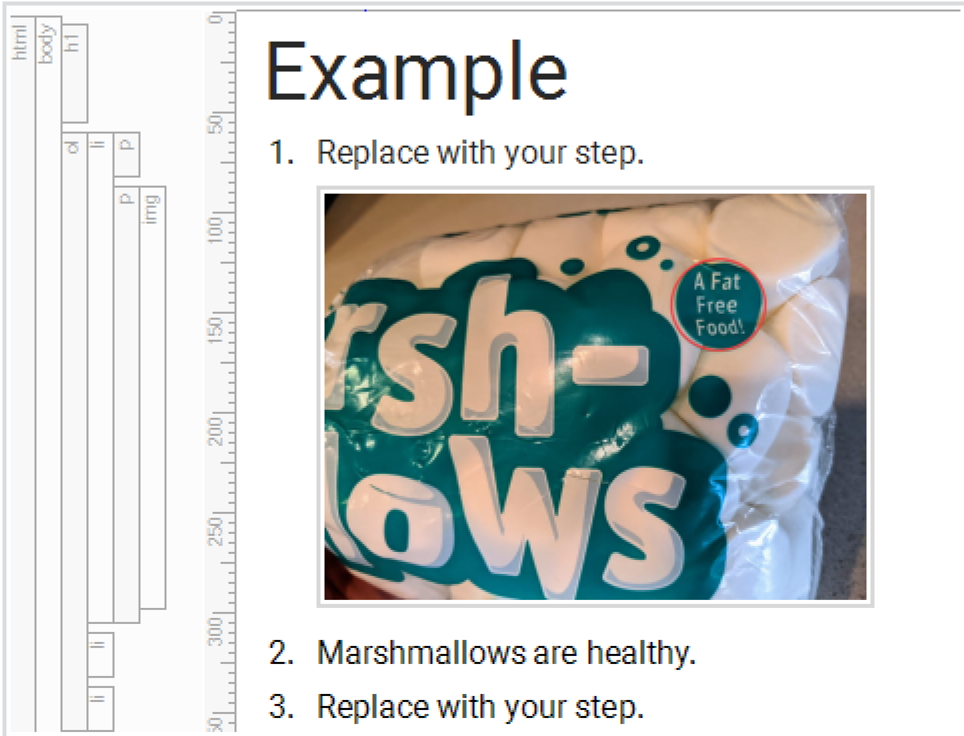
   c. Provide a brief description of the image in the **Screen Tip** and **Alternate Text** fields.

   d. Select **OK**.

   e. Select the image, then select the **Styles** panel and apply the style **img.ImgBorder-90-percent**.

      > **Note**: Occasionally, an image may appear too large in the output. If this occurs, you may need to apply a different border image style, such as img.ImgBorder-35-percent.

      The image proudly displays without a step number.



> **Note**: Occasionally, an image may appear too large in the output. If this occurs, you may need to apply a different border image style, such as img.ImgBorder-35-percent.

For more information, see the Flare Help topic Working with Paragraph and Simple List Items.

5.4 Lists

Lists are useful for providing content that's easy to scan. In addition, procedures wouldn't be procedures without numbered lists.

This topic provides conceptual information about lists, as well as procedures for working with lists.

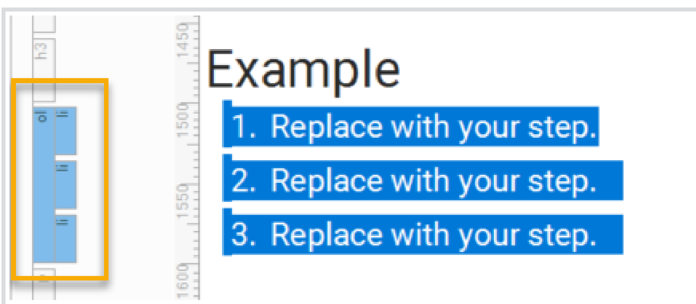
The MadCap Flare Help provides excellent information on lists. If you are new to Flare, read the topic Creating Numbered and Bulleted Lists.

In addition, see the related topics Inserting an Image into an Ordered List and Create a Procedure Topic.

This topic includes the following:

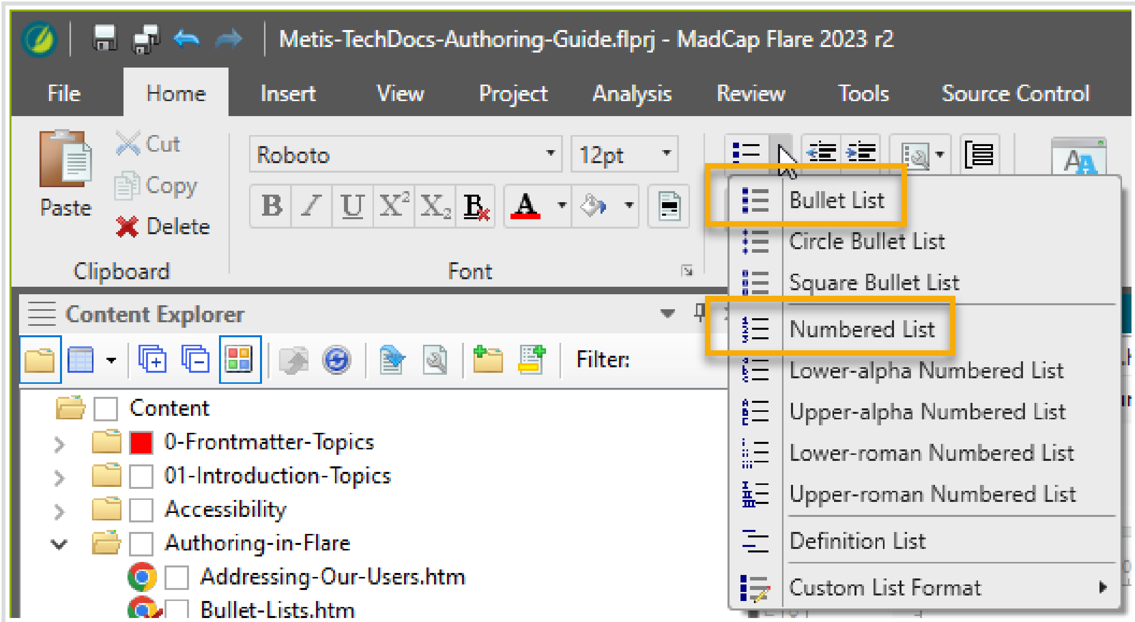
5.4.1 HTML List Elements

In HTML, a numbered list is represented by the tag <ol> for *ordered list*. A bullet list is represented by the tag <ul> for *unordered list*. When you insert a list into a topic, Flare displays an <ol> or <ul> tag bar to the left of the list. Each list item is represented by an <li> tag bar.



5.4.2 Generic Lists

The Flare Help recommends using the generic **Bullet List** and **Numbered List** options shown in the following image.



> **Important!** Only use the generic options **Bullet List** and **Numbered List** when working in the CD Help project. The CD Flare project has been set up to format lists appropriately if you follow the steps in <**add link here on tabbing to add sub-items**>.

5.4.3 List Levels

With both ordered lists and bulleted lists, the Tab key lets you move to a second or third list level. The Backspace key or the **Outdent** button  let you move up a level.

Ordered List Levels

The CD Help project is set up to automatically handle styles for up to three levels in ordered lists. The first list level uses numerals (1, 2, 3, etc.). The second list level uses lower alpha characters (a, b, c, etc.). The third list level uses lower roman characters (i, ii, iii, etc.). Rewrite steps to avoid going further than three levels.

Change List Levels

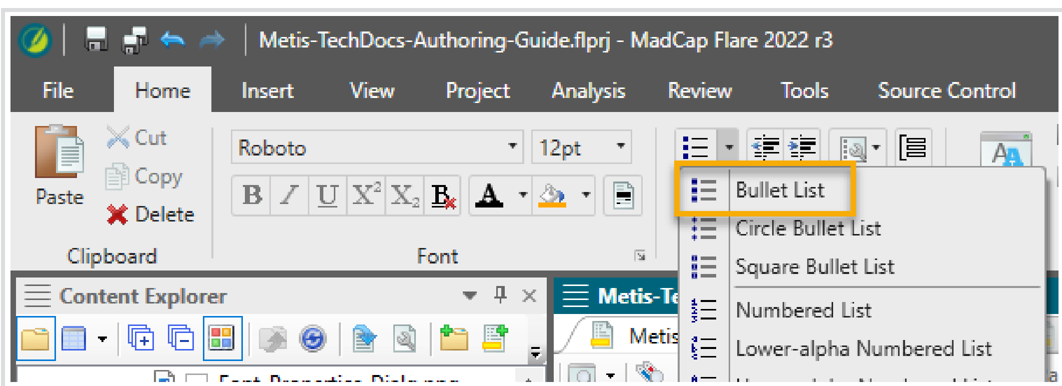
When you're working on a list and you press the Enter key, Flare adds a new line that continues the sequence.

You can change list levels as follows:

- To move to the second level of the list (lower alpha), press Tab. To add another lower alpha step, press Enter.
- To move to the third level of the list (lower roman), press Tab.

> **Note**: Flare won't let you add a third level with the Tab key if you haven't entered text at the second level.

- To move up a level, press Backspace or select the outdent button  in the **Paragraph** area of the **Home** ribbon.
- To change a sub-step to a bullet list item, select the generic **Bullet List** option from the list menu on the **Home** ribbon.



Fix the Numbering in an Interrupted List

If things get messy with numbering, and you can't get back to the highest level of the list, Flare provides the following workaround:

1. On a paragraph line immediately following the list, select the **Numbered List** option on the **Home** ribbon. Flare adds a new <ol> with a step **1**.

2. Right-click the <ol> tag bar that corresponds to the new list, and select **Continue Sequence**.

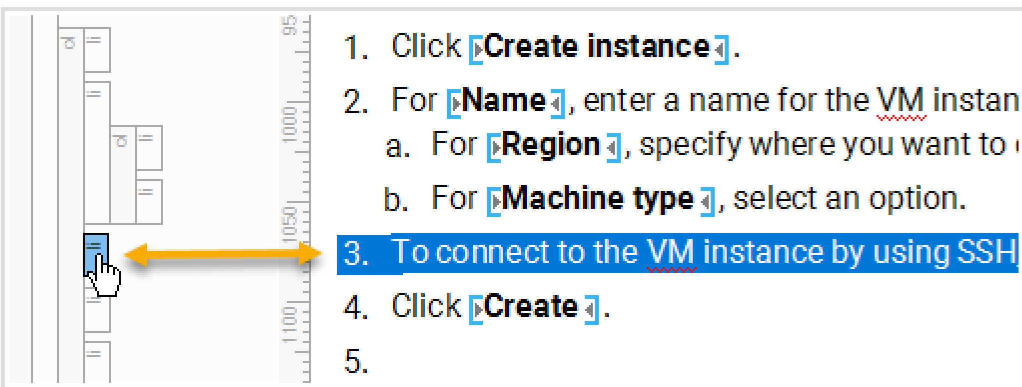
   Step **1** updates to the appropriate number where the previous numbered list left off.

> **Note**: You can also use the **List Start Number** option, but the number you set will not update if you insert a step before it.

Drag List Items to Reorder Steps

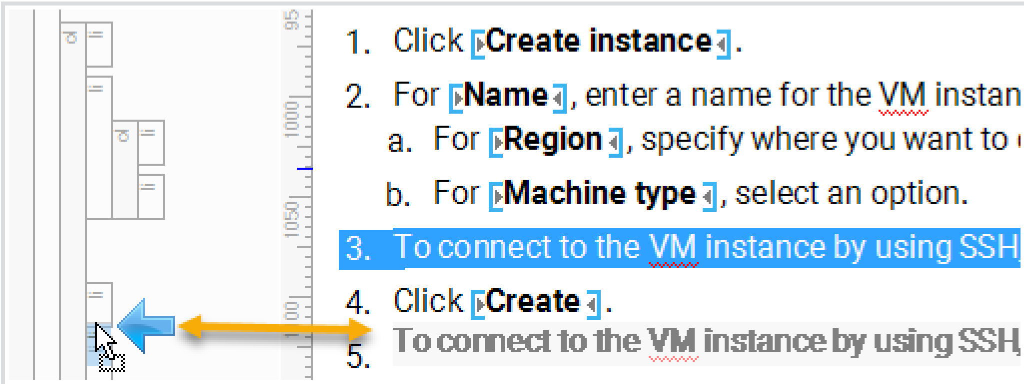
You can drag tag bars to reorder steps in a procedure as follows:

1. Click and hold down the left mouse button on the <li> tag bar that corresponds to the list item you want to move.

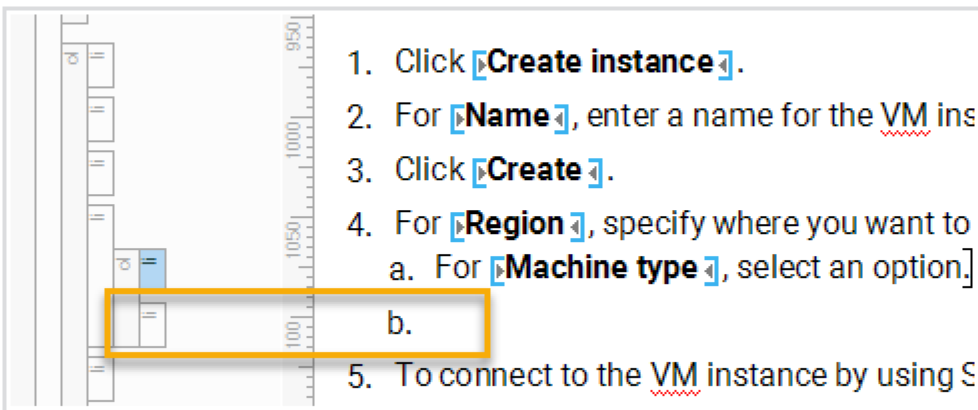


2. Drag the <li> tag bar up or down in the <li> tag bars, depending on where you want to move the list item.

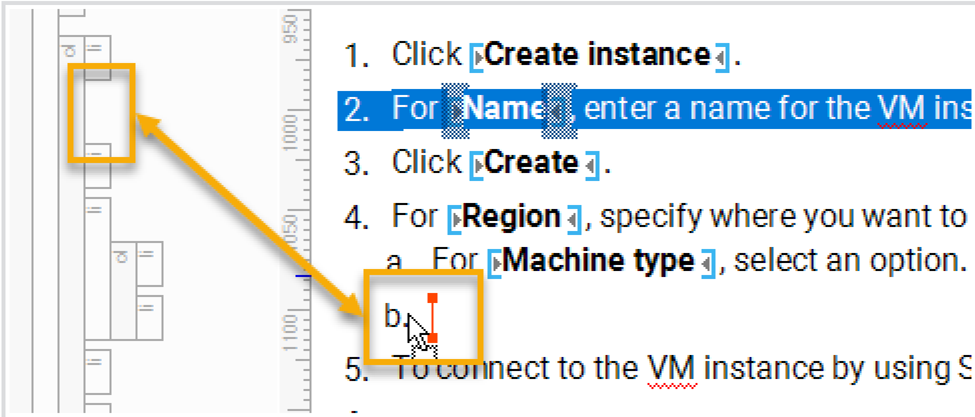
   As you drag the <li> tag bar up or down, the corresponding step moves with the <li> tag bar. An arrow appears whenever you drag the <li> tag bar to a place where you can place the step.



3. When the step is where you want to place it, release the left mouse button.
4. To drag a step to make it a second level step:
   a. Press Enter to add a sub-step where you want to move the step.



   b. Click and hold down the left mouse button on the <li> tag bar that corresponds to the list item you want to move.
   c. Drag the <li> tag bar into the XML Editor to the sub-step you added. For example, if you added sub-step b as in the above image, drag the <li> tag bar so that your cursor is directly after sub-step b.
   d. When a red line appears after the sub-step, release the left mouse button.



The step now appears as a sub-step in the second level list.

5.4.4 Bullet Lists

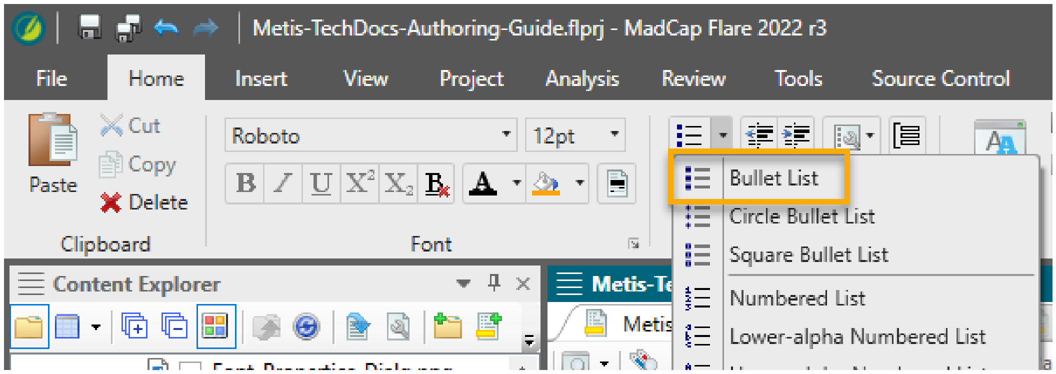
Use bullet lists for two or more ideas that have something in common that don't need to be in a particular order. Review the Microsoft Style Guide for bullet lists.

Although the items in a bullet list have something in common, each item is independent and doesn't rely on, or build upon, the others. In other words, a reader should be able to read the sentence that introduces the bullet list, then skip to any item in the list and have it make sense.

<p class="Rant">Bullet lists aren't the Swiss Army knives of technical writing and shouldn't list random, unrelated ideas.</p>

The following bullet list summarizes the guidelines for bullet lists:

- Use a sentence or phrase ending with a colon (:) to introduce a bullet list.
- Select the first menu option, **Bullet List**, on the **Home** ribbon.



- Use sentence case.
- Use either all complete sentences or all fragments in a bullet list.
- Use a period at the end of complete sentences, even if the complete sentence is very short.

> The following occurs after the task rejection:
>
> - The task is moved to the **Available To Me** tab.
> - The **Reason for rejection** comment is required in the **Comments** tab on the **Task Details** page for the subject task.
> - The **History** tab on the **Task Details** page indicates that the owner has been reassigned to "Unassigned".

- Omit periods at the end of list items that are fragments.

> **Edit a Form**
>
> Depending on your role, jurisdiction, and permissions, you can edit the following types of forms:
>
> - National forms
> - State forms
> - Service center forms

Suboptimal Bullet Lists

The following image is an example of a suboptimal bullet list.

3.2.1. Condition Groups for Sections

On the **Display Group** page, in the **Section Name** area, select the **Conditions** tab.

- Select the **Overall Operator** drop-down arrow and select the operator the section or question will use to process the results coming from more than one conditional group.
  The default is **AND**.
  - All answers in a conditional group are tested using the same operator.
- Select **ADD CONDITIONAL GROUP** to add a conditional group to the section or question.
- Every condition you create must be assigned to a conditional group.
- Each conditional group is configured using either an **AND** or an **OR** operator to process the choices contained in the conditional group.

In the above image, some of the content is procedural, while some of it is conceptual. The procedural information belongs in a procedure topic. The conceptual information should be organized into one or more paragraphs that coherently link the ideas together.

5 Source Control

The topics in this section describe how to perform activities involving source control for Flare projects:

5.5 Request a Repository or Access to a Repository

The easiest way to request a repository or to request access to a repository is to clone a Jira request and edit the request as needed. You can use DEVOPS-16114 to request a repository, and DEVOPS-17215 to request access to a repository. If you are unable to clone these Jira stories, use the steps below to create a request.

1. In a web browser, open Jira.
2. In the menu at the top of the page, select **Create**.
   The **Create Issue** dialog opens.
3. In the **Project** field, select **NRCS Operations (DEVOPS)**.
4. For **Issue Type**, select **Story**.
5. In the **Summary** field, enter **Create Bitbucket Repo for Application Help Content** or **Write Access and Pull Request Approval Needed for Bitbucket Repo**.
6. Under **Component/s** select **NRCS ART Service Request**, and then select **NRCS OPS Requests**.
7. In the **Description** field, use one of the following options and update the provided boilerplate as needed:

- ***For access requests***:

  Please add the following personnel to this repository.

  Repository Name: <Repository-Name>

  Users needing branch write access and pull request approval:

  - <Philma Payne>
  - <Howard Aryou>
- ***For repository requests:***

  The <Application-Name> application requires a repository for MadCap Flare help documentation. This repository will not need any pipeline setup.

  Repository Name: <Application-Name>-Help

  Repository Description: Help source content for <Application-Name>

  Users that need branch write access and pull request approval:

  - <Philma Payne>
  - <Howard Aryou>

8. Select the appropriate **Program Increment**.
9. Select **Save**.

5.6 Setting Up Source Control

The following topics describe how to establish source control for a Flare project.

5.6.1 Source Control Prerequisites

The following items are prerequisites:

- You have a Bitbucket account.
- You know you're Crowd user name and password. Reset your Crowd password if necessary.
- You have Git Bash installed.
- A Bitbucket repository exists for your Flare project and you have branch write access and pull request approval for the repository.

5.6.2 Set Your User Name in Git

Git uses a user name to associate commits with an identity.

1. Open Git Bash:
   a. In the Windows search field, enter `Git Bash`.
   b. Right-click on Git Bash in the search results and select **Open file location**.
   c. In Windows Explorer, right-click on Git Bash and select **Run with Elevated Privileges**.
2. Enter the following commands, substituting your name and email address:
   `git config --global user.name "Joe Schmoe"`
   `git config --global user.email "joseph.schmoe@usda.gov"`
3. Verify what you just done:
   git config --list

5.6.3 Set Up SSH Keys

SSH keys establish a secure connection from your local machine to the Bitbucket repository.

1. Open Windows Command Prompt:
   a. In the Windows search, enter `cmd`.
   b. Right-click on Command Prompt, and select **Open file location**.
   c. In Windows Exploder, right-click on **Command Prompt**, and select **Run with Elevated Privileges**.
2. Check for existing SSH keys:
   a. In the Command Prompt, enter the following:
      `cd ..\..\Users\<Joseph.Schmoe>\.ssh`

If you see a response `No such file or directory`, you don’t have existing SSH keys. Go to step 3.

b. Check for existing keys:

```
dir id_*
```

If you have existing keys, go to step 4.

3. Generate SSH keys:

a. In the Command Prompt, enter the following:

```
ssh-keygen -t rsa -b 4096 -C "your_email@example.com"
```

> **Note**: Associating the key with your email address helps you to identify the key later.

You'll see a response like this:

```
C:\windows\system32>ssh-keygen -t rsa -b 4096 -C
joseph.schmoe@usda.govGenerating public/private rsa key
pair.Enter file in which to save the key
(C:\Users\joseph.schmoe/.ssh/id_rsa):
```

b. Press <Enter> to accept the default location and file name. If the .ssh directory doesn't exist, the system creates one for you.

c. Press <Enter> when prompted for a passphrase, then press <Enter> again when prompted to enter the same passphrase.

The whole interaction looks like this:

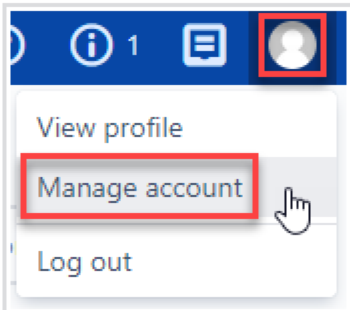
```
C:\windows\system32>ssh-keygen -t rsa -b 4096 -C
joseph.schmoe@usda.govGenerating public/private rsa key
pair.Enter file in which to save the key
(C:\Users\joseph.schmoe/.ssh/id_rsa):Enter passphrase (empty for
no passphrase):Enter same passphrase again:Your identification
has been saved in C:\Users\joseph.schmoe/.ssh/id_rsa.Your public
key has been saved in C:\Users\joseph.schmoe/.ssh/id_rsa.pub.The
key fingerprint
is:SHA256:DMxdjsH3S2CxHoRf0eEfxmDXIwyKWurltJTZHs6LKb4
joseph.schmoe@usda.gov
```

4. Copy the public SSH key to the clipboard:

```
clip < rsa_id.pub
```

> **Note**: If you already had existing keys, change the directory to your .ssh directory first: `cd <joseph.schmoe>/.ssh`

5. In Bitbucket:
   a. Select the user icon, and then select **Manage account**.

   b. Select **SSH keys**, and then click **Add key**.
   c. Paste the key from C:\Users\joseph.schmoe/.ssh/id_rsa into the text box.

   > **Note**: When you paste your key into Bitbucket, if it doesn't start with ssh-rsa and end with joseph.schmoe@usda.gov, use the command clip < rsa_id.pub in the Windows command prompt to copy the key.

   d. Click **Add key**.

5.6.4 Binding Your Project to Bitbucket

In the Flare world, *binding a project* means establishing source control for a project. Currently, we use Bitbucket for source control. There are multiple ways to bind your project to Bitbucket. The steps that follow provide one method.

At a high level, you clone the Bitbucket repository to your local system, copy your Flare project files to the cloned directory, and then push your files to Bitbucket.

> **Note**: The following steps assume you're using C:/My Projects for MadCap Flare projects.

Bind Your Project to Bitbucket

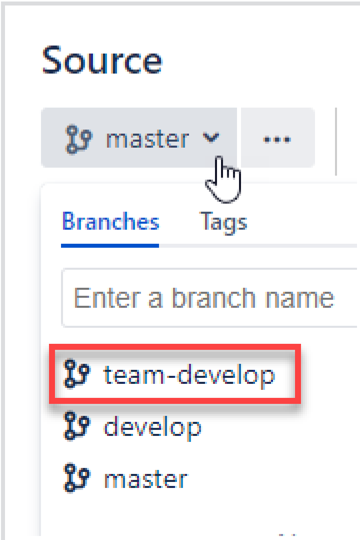
> **Important!** Use the following steps only if your project is *not* already in Bitbucket. If you're trying to clone a project that is already in Bitbucket, follow the steps inClone a Flare Project in Bitbucket (on page 63).

1. Open Git Bash and change directories to your My Projects folder:
   `cd /c/My\ Projects/`
2. Clone the Bitbucket repository:
   `git clone https://bitbucket.fpac.usda.gov/scm/nrcsapps/<repo-name>.git`

   > **Note**: If cloning fails with an error that says SSL certificate problem: self signed certificate in certificate chain, enter the following syntax to clone the repository:
   > `git -c http.sslVerify=false clone <repository-name>`

3. Change directories to the cloned project folder:
   `cd /c/My\ Projects/<cloned-project>`
4. Create a team-develop branch and switch to the team-develop branch:
   `git checkout -b team-develop`

   If you open or refresh Bitbucket, a **team-develop** branch now appears in the **Branches** menu.



> **Note**: You don't have to use the name "team-develop," however, you can't use "develop" for your working branch. Branches named "develop" and "master" are locked down, and you can't push changes to them without doing a pull request.

5. Use a text editor to open the Git ignore file in My Projects/<project-name>/<cloned-project>. Replace the text in the ignore file with the text shown in the topic Git Ignore File (on page 62).
6. Copy all of your local project files to your cloned project folder:
```
cp -a /c/My\ Projects/<local-project>/* /c/My\ Projects/<cloned-project>/
```
7. Stage your files to be committed:
```
git add --all
```
8. Commit your files adding an "initial commit" comment:
```
git commit -m "initial commit of files"
```
9. Push your files:
```
git push
```
When the push completes, you can click the **Commits** icon in Bitbucket to verify your commit.

5.6.5 Git Ignore File

Depending on the method you use to establish source control for a Flare project, you may need to create or edit an ignore file. An ignore file specifies files that don't need to be tracked by source control. For more information, see the MC Flare help topic Files and Folders.

To create an ignore file, copy the following text into a text editor and save the file as .gitignore:

```
bin
target
node_modules
.settings
.metadata
.sonarlint
.vscode
.classpath
.project
################################################################################
#
# Placing a # in front of any of the items listed below (e.g., #analyzer)
will  #
# allow these items to be pushed to your Git repository. Keep in mind the
more  #
# files you push to your repository, the more your performance may be affected.
#
################################################################################
#
/Output
/Analyzer
/FileSync
/Project/Users
/.svn
**/Thumbs.db
################################################################################
#
# Add items to be ignored
below                                                 #
################################################################################
#
```

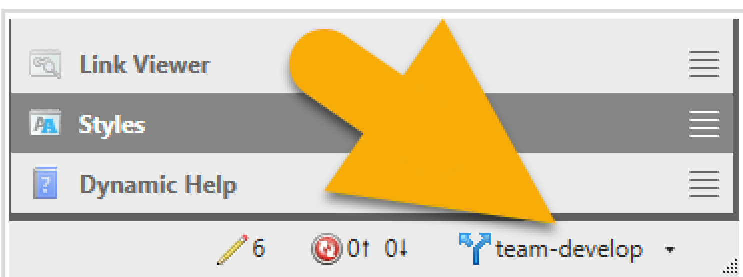
5.7 Clone a Flare Project in Bitbucket

Use the following steps to clone a project that already exists in Bitbucket.

1. Open Git Bash and change directories to your My Projects folder:
   `cd /c/My\ Projects/`
2. Clone the Bitbucket repository:
   `git clone https://bitbucket.fpac.usda.gov/scm/nrcsapps/<repo-name>.git`

   > **Note**: If cloning fails with an error that says SSL certificate problem: self signed certificate in certificate chain, enter the following syntax to clone the repository:
   > `git -c http.sslVerify=false clone <repository-name>`

3. Change directories to the cloned project folder:
   `cd /c/My\ Projects/<cloned-project>`
4. Switch to the team-develop branch:
   `git checkout team-develop`
5. In Windows Explorer, browse to your My Projects folder, and open the Flare project you just cloned.
6. In the lower right corner in the Flare project, verify that the **team-develop** branch displays.



5.8 Release a New Version of the CD Help

The CD Help PDF is included in the build of the CD application. On the Friday before code complete (DoD) for a release, Metis Tech Docs builds an updated version of the CD Help PDF.

To get the updated CD Help PDF into the build, two things need to happen:

1. Metis Tech Docs pushes the updated CD Help PDF to the Bitbucket repository **CD-CDSI-Help**.
2. A developer triggers a build of CDSIResources.

When CDSIResources builds, it pulls in the CD Help PDF from the CD-CDSI-Help repository.

The following topics describe the tasks Metis Tech Docs performs to get the CD Help into the build:

5.8.1 Prerequisites for Releasing the CD Help

The following items are prerequisites:

- You have a Bitbucket account.
- You know you're Crowd user name and password. Reset your Crowd password if necessary.
- You have Git Bash installed.

- You have permission to commit directly to the Develop branch of the CD-CDSI-Help repository without a pull request.

5.8.2 Update Variables and Build the Help PDF

Before you build the PDF that you will push to the CD-CDSI-Help repo, use the following steps to update variables so that the version and date information in the PDF are correct.

1. Open the CD-Help Flare project.
2. On the **Source Control** tab, select **Pull** to ensure your local repository is up to date.
3. In the CD Flare project:
   a. Navigate to the **Project Organizer**. Expand the **Targets** folder and double-click the target **CD-Help-PDF (Primary)**.
   b. Select the **Variables** tab and then select **MyVariables**.
   c. Change **UpdatedDate** to the current date.
   d. Change the **Version** to the version number for the current release.
4. Save your changes.
5. Build the target **CD-Help-PDF (Primary)**.
6. When the CD Help PDF opens:
   a. Save the PDF as ***Conservation_Desktop_Comprehensive_Users_Manual***.
   b. Save the PDF either to your desktop or to a location you'll remember.
7. If you haven't already cloned the CD-CDSI-Help repository, continue to Clone the CD-CDSI-Help Repository. If you already cloned the CD-CDSI-Help repository, go to Push the CD Help PDF to the CD-CDSI-Help Repo.

5.8.3 Clone the CD-CDSI-Help Repository

Use the following steps *only* if you haven't already cloned the CD-CDSI-Help repo. If you have already cloned the CD-CDSI-Help repo, continue to the next section.

1. Use Windows Explorer to create a folder called **CD-CDSI-Help**.
2. Right-click on the new **CD-CDSI-Help** folder and select **Git Bash Here**.
3. Enter the following command:

   `git clone https://bitbucket.fpac.usda.gov/scm/nrcsapps/cd-cdsi-help.git`

   > **Note**: If cloning fails with an error that says SSL certificate problem: self signed certificate in certificate chain, enter the following syntax to clone the repository:
   > `git -c http.sslVerify=false clone <repository-name>`

   The cloning operation begins and may take several minutes to complete.
4. When the cloning operation completes, continue to the section Push the CD Help PDF to the CD-CDSI-Help Repo.

5.8.4 Push the CD Help PDF

Use the following steps *only* if you cloned the CD-CDSI-Help repo previously. If you *haven't* already cloned the CD-CDSI-Help repo, use the steps in Clone the CD-CDSI-Help Repository before continuing.

1. Use Windows Explorer to navigate to your **CD-CDSI-Help** folder.
2. Right-click on your **CD-CDSI-Help** folder, and select **Open Git Bash Here**.
3. In Git Bash, enter the following command to check the status of the working directory and the staging area:

   `git status`
4. Pull changes from the remote repository:`git pull --all`

   > **Note**: If you get an error that says SSL certificate problem: self signed certificate in certificate chain, enter the following command:
   > `git -c http.sslVerify=false pull`
5. Use Windows Explorer to copy the CD Help PDF you saved to your desktop to your **CD-CDSI-Help** folder.

6. In Git Bash, enter the following command to check the status of the working directory and the staging area:

   `git status`

7. Stage your files to be committed:

   `git add .`

8. Commit your files adding a comment in quotes:

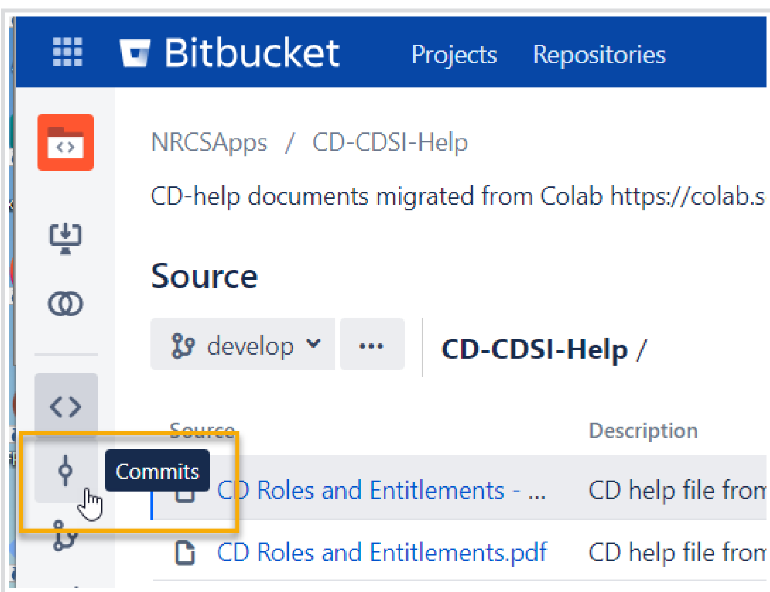
   `git commit -m "your comment here"`

   > **Note**: The best practice is to indicate the version number and release name in your comment.

9. Use one of the following commands to push your changes to the remote repository. If you previously got an SSL certificate error, use the second option:

   - `git push`
   - `git -c http.sslVerify=false push`

10. Go to the [CD-CDSI-Help repository in Bitbucket]() and select the **Commits** icon. Confirm that the new PDF was pushed to the repository.



11. To ensure that the CD Help PDF gets into the build, contact Robert Waasbergen and ask him to kindly trigger a build of CDSIResources for Dev2.

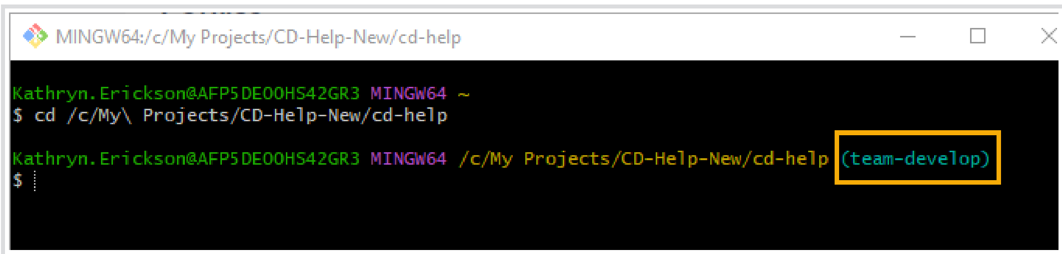
5.8.5 Create a Branch for the Release

The last release activity is to create a branch for the release. If there is ever a need to update the help for a specific release, the branch you create will give you a way to do so.

1. Use Windows Explorer to navigate to your **cd-help** folder. Right-click on the **cd-help** folder and select **Open Git Bash Here**.

   When Git Bash opens, you should see that you are on the **team-develop** branch.

   > **Note**: If you are not on the team-develop branch, use the command `git checkout team-develop` to switch to the team-develop branch. If you don't see a branch, verify that you're in the correct directory.

2. In Git Bash:

   a. Enter the following command to create a new branch for the current release:

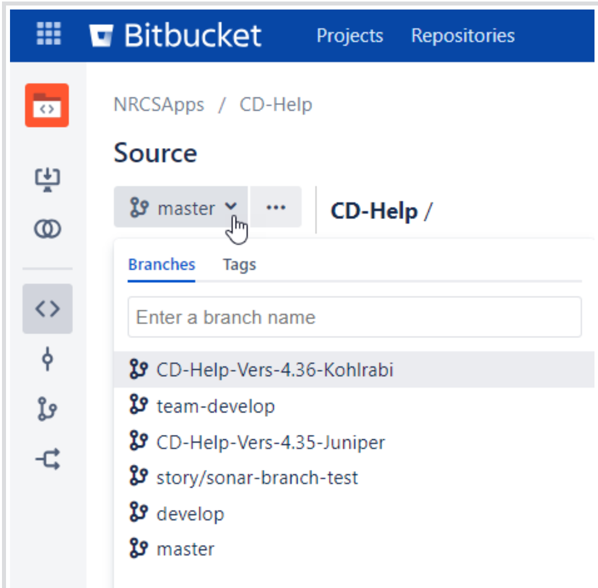
      `git checkout -b CD-Help-Vers-<#.##>-<Release-Name>`

      For example, the branch for the latest release is *CD-Help-Vers-4.36-Kohlrabi*. Use the version number that you used for the **Version** variable in Flare.

   b. Push the new branch to the repository:

      `git push -u origin CD-Help-Vers-<X.YZ>-<Release-Name>`

3. Go to the [CD-Help repository in Bitbucket] and verify that the new branch appears.



4. Switch back to the team-develop branch:

```
git checkout team-develop
```

6 Troubleshooting

The following topics provide procedures for troubleshooting some of the issues you may encounter:

6.1 Pushing to Bitbucket Fails with Error "Failed to Push Some Refs"

This topic provides troubleshooting for some of the scenarios where pushing to a repository fails with the error **error: failed to push some refs to '<https://bitbucket.fpac.usda.gov/scm/nrcsapps/<repo-name>.git>'**. This error can occur for several reasons, such as if you forgot to pull before you pushed. If this is the case, do a pull, and then push again.

The following sections cover other instances of this error:

6.1.1 Permissions

You may get the error **error: failed to push some refs to '<https://bitbucket.fpac.usda.gov/scm/nrcsapps/<repo-name>.git>'** if you don't have adequate permissions for the repository you're working with. For example, if you are trying to push the CD Help PDF to the CD-CDSI repository, your permissions must include write access to the ***develop*** branch. If necessary, create a Jira ticket to edit your permissions.

6.1.2 FLPRJ File Is Too Large

Each Flare project has a .flprj file. You can double-click a .flprj to open a Flare project. When you open a .flprj in Flare, you're not actually opening and editing that file directly. A .flprj file is an .xml file that stores certain configuration settings for the project. If you do something like bind the project to source control, or define a default stylesheet, an entry will appear in the the .flprj file. In addition, any time you import files into the project, the URL of the imported files and a timestamp are recorded in the .flprj.

The .flprj file is usually very small. However, projects that are linked to a global project containing many files may end up with a very large .flprj file. If the .flprj gets over 5MB and you try to push to Bitbucket, the push will be blocked and you'll get the error **error: failed to push some refs to '<https://bitbucket.fpac.usda.gov/scm/nrcsapps/<repo-name>.git>'**. The error message will also indicate that the .flprj was blocked due to file size.

Reduce the Size of FLPRJ File

1. Copy the .flprj file and paste it on your desktop. This is a backup that you will probably not need.
2. Browse to your **My Projects** folder, and open the .flprj in Notepad++ or a similar text editing tool.
3. Locate the opening tag **<CollectedSourceFiles>** and delete all lines until you reach the closing tag **</CollectedSourceFiles>**. Ensure that you do not delete the opening and closing tags **<CollectedSourceFiles>** and **</CollectedSourceFiles>**.
4. Locate the opening tag **<SkippedSourceFiles>** and delete all lines until you reach the closing tag **</SkippedSourceFiles>**. Ensure that you do not delete the opening and closing tags **<SkippedSourceFiles>** and **</SkippedSourceFiles>**.
5. Locate the opening tag **<GeneratedFiles>** and delete all lines until you reach the closing tag **</GeneratedFiles>**. Ensure that you do not delete the opening and closing tags **<GeneratedFiles>** and **</GeneratedFiles>**.

6. Save your changes.
7. Double-click the .flprj file and verify that it opens.

6.2 Flare Has Become Slow

If Flare is being sluggish, for example, if there's a lag when you're typing in the XML Editor, or the Style Window is slow to update with available styles, the following steps from Flare Support may resolve the issue.

1. Save your opened files and close Flare.
2. Clear Windows Registry for Flare:
   a. Type **regedit** in the Windows search bar, and then select **Registry Editor**.
   b. In **Registry Editor**, browse to **HKEY_CURRENT_USER/Software/MadCap Software**, and locate the **Flare** folder.
   c. Right-click on the **Flare** folder, and select **Export** to create a backup of the folder.
   d. In the **Export Registry File** dialog, select a location for the backup. Name the file **Flare-Reg-Edit-Backup**.
   e. In **Registry Editor**, right-click the **Flare** folder and select **Delete**.
3. Clear Flare's AppData:
   a. Navigate to **C:\Users\<your.name>\AppData\Roaming\MadCap Software**.

      > **Note**: If you don't see an AppData folder, follow the steps in View Hidden Files in Windows Explorer, and then resume this step.

   b. Copy and paste the **Flare** folder to another location, and append the copy with the word **Backup**.
   c. Delete the **Flare** folder in **C:\Users\<your.name>\AppData\Roaming\MadCap Software**.
4. Cross your fingers and reopen Flare to test the behavior.

6.2.1 View Hidden Files in Windows Explorer

1. Open a Windows File Explorer.
2. Select **View**, select **Options**, and then select **Change folder and search options**.
3. Select the **View** tab, and in **Advanced settings**, select **Show hidden files, folders, and drives**.
4. Select **OK**.

6.3 Pull, Push, or Sync Isn't Working in Flare

For reasons that are unclear, the **Push**, **Pull**, and **Synchronize** buttons on the **Source Control** ribbon may stop working in your Flare project. If this happens, Flare won't allow you to commit changes or update your local repository with changes others have made. Before you call Flare support, you can use Git Bash to synchronize your files.

1. In Windows Explorer, navigate to and right-click on the folder of the problematic Flare project, and then select **Git Bash Here**.
2. In Git Bash, enter the following command to check the status of the working directory and the staging area:

   `git status`

   > **Note**: The git status command lets you see which changes have been staged, which haven't, and which files aren't being tracked by Git. If you have uncommitted changes, they will show up here.

3. If you have uncommitted changes, stage your files to be committed:

   `git add --all`

4. Commit your files adding a comment in quotes:

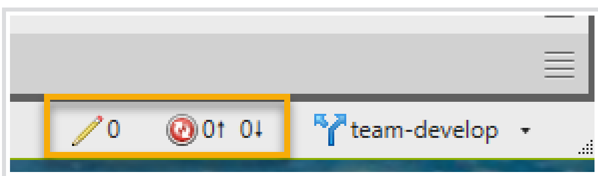
   `git commit -m "your comment here"`

5. Pull changes from the remote repository:

   `git pull --all`

   > **Note**: If you get an error that says SSL certificate problem: self signed certificate in certificate chain, enter the following command:
   > `git -c http.sslVerify=false pull`

6. Use one of the following commands to push your changes to the remote repository. If you previously got an SSL certificate error, use the second option:
   - `git push`
   - `git -c http.sslVerify=false push`
7. If you were able to complete the previous steps, cross your fingers and look at your Flare project. In the lower right corner, Flare should show that there are no pending changes or changes to be synchronized.

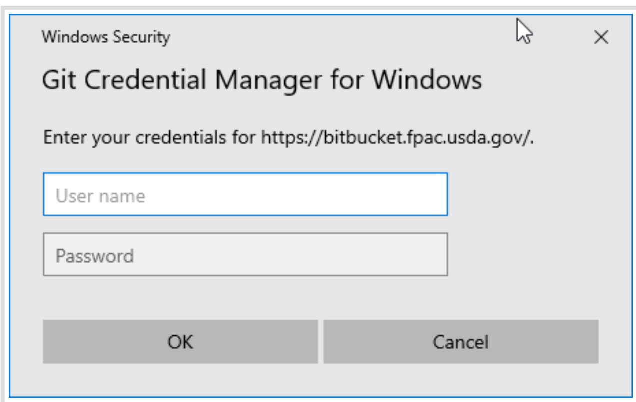
8. Make a small change and then try to synchronize.
9. If the **Push**, **Pull**, and **Synchronize** buttons are still not working, but you have no uncommitted changes (other than the small change you made in the previous step), do the following:
   a. Close Git Bash and Flare if they're still open.
   b. Rename the folder of the Flare project that's borked by appending it with the word "OLD," for example, CD-Help-OLD.
   c. Follow the steps in Clone a Flare Project to clone the project again.

6.4 Wrong Password in Credential Manager

The Windows Credential Manager stores authentication credentials for signing in to websites, connected applications, and networks.

When you're working in Flare, if a Windows Security dialog prompts you to enter your credentials for https://bitbucket.fpac.usda.gov/:

- Enter your firstname.lastname for the **User name**.
- Enter your eight-digit PIV for the **Password**.



Important! Do not enter your Crowd credentials in the Credential Manager.

The good thing about Credential Manager is that it keeps you from having to log in to Bitbucket every time you do something with source control. The bad thing about Credential Manager is that if you fat-finger your credentials, Credential Manager will store them that way, and then your authentication attempts will fail. In addition, it's not apparent that you fat-fingered your password, because Credential Manager hides your password.

6.4.1 Update Git Credentials in Credential Manager

1. In the Windows search field, start typing `Credential` and select **Credential Manager** when it displays.
2. Select **Windows Credentials**.
3. Under **Generic Credentials**, select **git:https://bitbucket.fpac.usda.gov**, and then select **Edit**.
4. In the **Edit Generic credential** window, update your credentials and then select **Save**.

6.5 Application Won't Open and Asks for License Key

Many applications that we use have licenses that IT manages with policies (e.g., Snagit). If an application you normally use won't open and asks you to enter a license key, use the following steps to force a policy update.

1. Open the Windows Command Line utility.
2. Enter the following command:

```
$gpupdate /force
```

After you force the policy update, the application should open and behave as expected.

7 Flare Tips and Tricks

The following topics provide pro tips for using Flare and explain some Flare features that aren't immediately obvious:

7.1 Top Ten Flare Features for Beginners

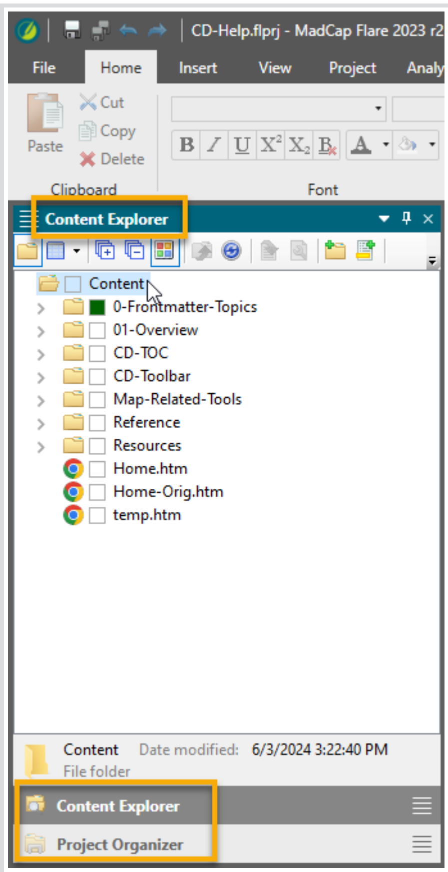
This topic provides an introduction to some basic Flare features. When possible, this topic provides links to additional resources.

This topic includes the following:

Content Explorer and Project Organizer

The Content Explorer is where you'll spend most of your time working in Flare. Topic files and certain non-topic content files are organized in the Content Explorer. For more information on how content is organized in the Content Explorer, see the topic CD Flare Project Folder Structure.

The Resources folder in the Content Explorer is where certain non-topic files live. The files you will interact with the most in the Resources folder are images and snippets.



The Project Organizer is where project-related files are stored. The files you will interact with the most in the Project Organizer are TOCs and targets.

Dragging Items from the Content Explorer

You can drag items from the Content Explorer to add them to a topic. For example, you can drag an image from the Content Explorer into a topic you have open. Or, you can add an xref by dragging a topic file into a topic you have open.

Bold Font

The **B** (bold) button, as well as other buttons in the **Font** area of the **Home** ribbon, are disabled in the CD Flare project. These buttons apply inline styles rather than applying styles defined in the stylesheet. It's a best practice to avoid inline styles, because if you want to change the appearance of your content, you have to edit every single instance of an inline style attribute, rather than simply modifying the stylesheet.

☆ To apply bold font weight, highlight the text and use the style **span.UIText** in the **Styles** window.

In addition, see the following related topics:

- Character Styles
- Formatting UI Elements
- Create Keyboard Shortcuts in Flare

Carriage Returns

To start a new line without starting a new paragraph (carriage return), use Shift+Enter.

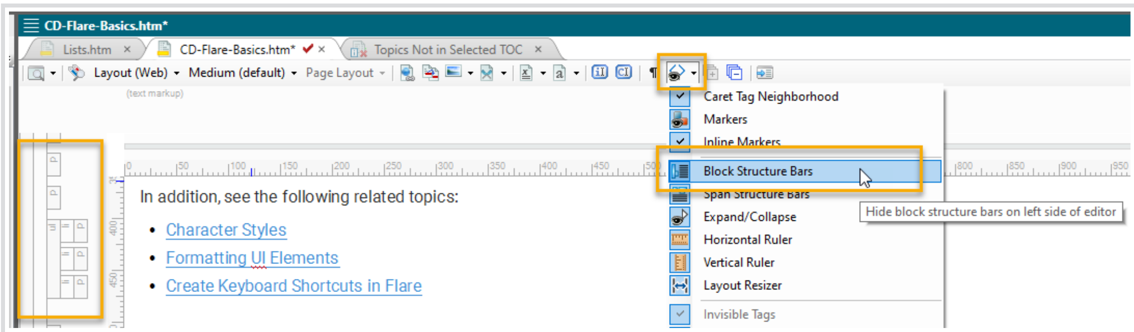
Changing List Levels

You can use the Tab key to move to a new list level. See the topic Change List Levels.

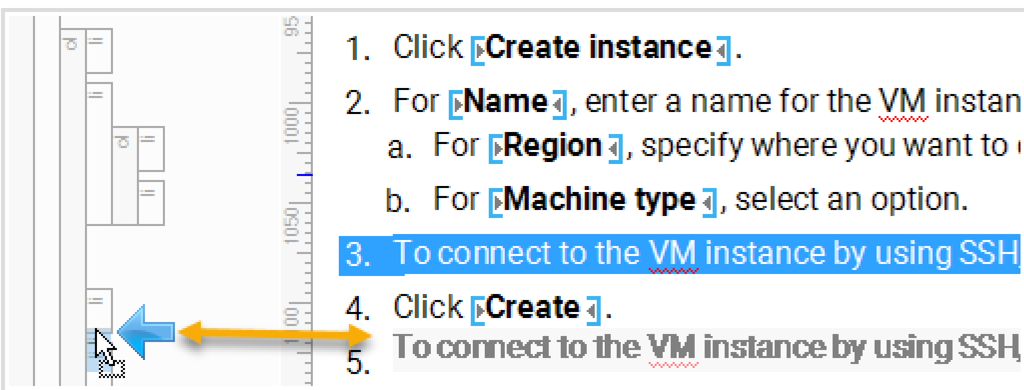
Dragging Tag Bars

In the XML Editor, tag bars display on the left-hand side.

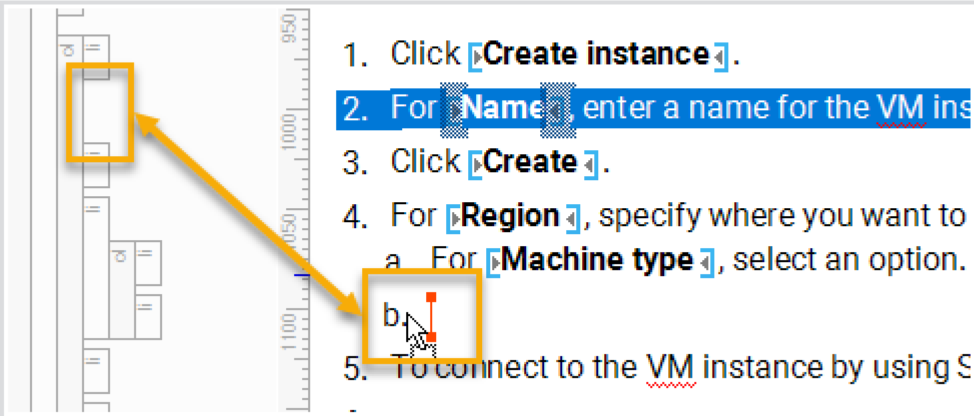
If you don't see tag bars, select the **Show tags** button, and then select **Block Stucture Bars**.



You can drag tag bars to move content in a topic. For example, you can drag a tag bar up or down in the tag bars to reorder paragraphs or list items.



You can drag a tag bar into the XML Editor. When a red line appears, you can release the left mouse button to move the item to the location of the red line.



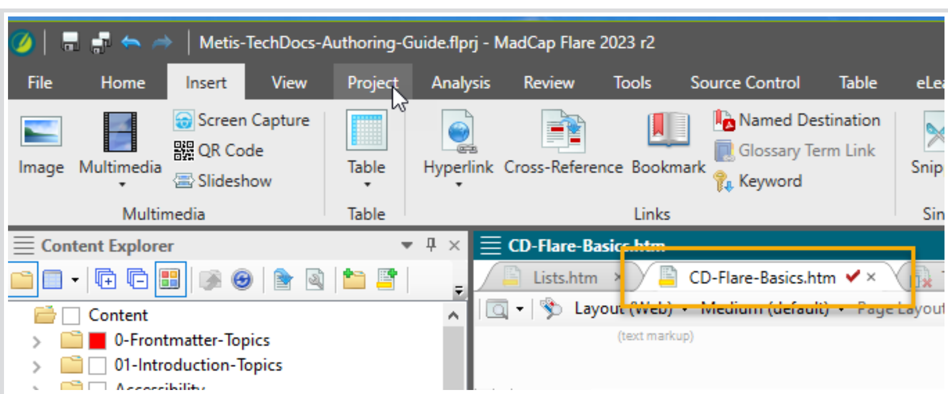
For more information, see the topic Drag List Items to Reorder Steps.

Link Viewer

The **Link Viewer** lets you see where a file is used. To open the Link Viewer, select the **View** ribbon, and then select **Link Viewer**.

Locating Files

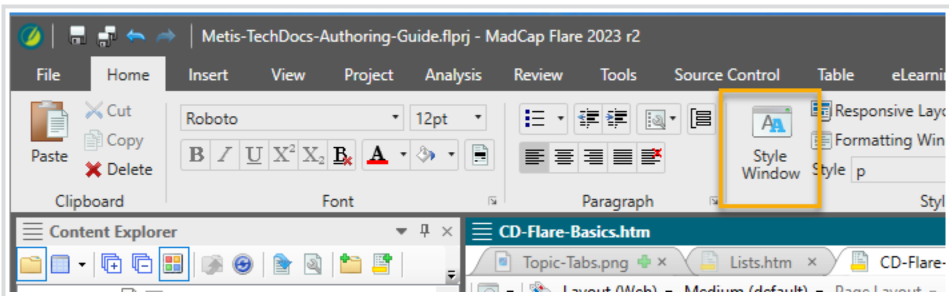
If you have a topic open, you can locate where the topic file is stored. Right-click the topic's tab and select **Locate in Explorer**. The following image shows a topic's tab.



In addition, you can right-click on a topic in a table of contents (TOC) and select **Locate in Explorer**.

Pinning Styles

The **Styles** window lets you select from a list of available styles. To open the **Styles** window, select the **Style Window** button on the **Home** ribbon.



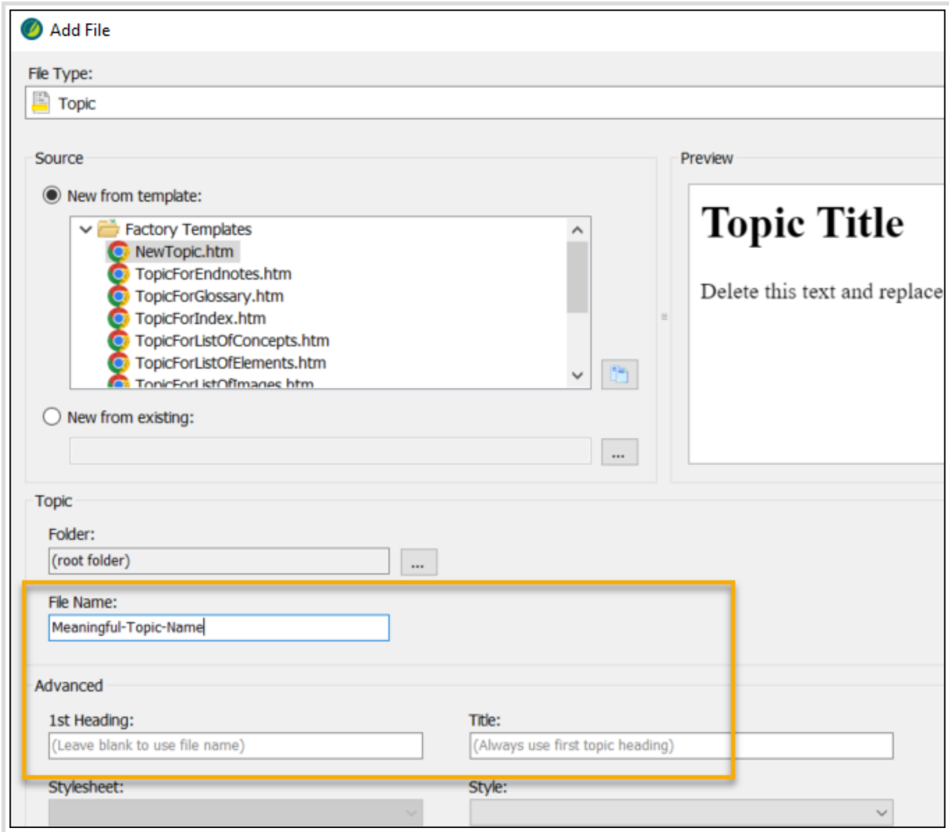
You can pin styles that you use frequently to make them appear at the top of the list in the **Styles** window. See the Flare Help topic Pinning Styles.

7.2 Topic File Names, Headings, and Titles

This topic explains Flare's default behavior with file names, first headings, and the titles of topic files.

If you're in a hurry, the main takeaway is that when you create a topic, make sure you populate the **1st heading** field in the **Add File** dialog, *exactly how you want it to appear*. Read on to understand why.

When you create a new topic, you enter a file name for the topic. By default, Flare uses the file name for the first heading in the topic, and for the title of the topic.



The guideline for topic file names is to separate words in the file name with hyphens (see File Names), as shown in the above image. However, if you leave the **1st Heading** field empty, the first heading and the topic title will use the hyphenated file name.

If a file name with hyphens is used for the first heading, you'll have to remove the hyphens later, since you won't want the heading to contain unnecessary hyphens. The following image shows an h1 heading with unnecessary hyphens. Not pretty.

Topic titles appear in the TOC, cross references, search results, and in the title bar of browser windows. If a file name with hyphens is used for the title, you'll have to remove the hyphens later from the title so that unnecessary hyphens don't appear. Cross references don't automatically update after you remove the hyphens, so you'll also have to reinsert cross references to get rid of the hyphens.

Therefore, the recommended practice when you create a topic is to *enter a first heading without hyphens*. You can copy and paste the file name, just make sure to remove any hyphens that aren't necessary for grammatical reasons.

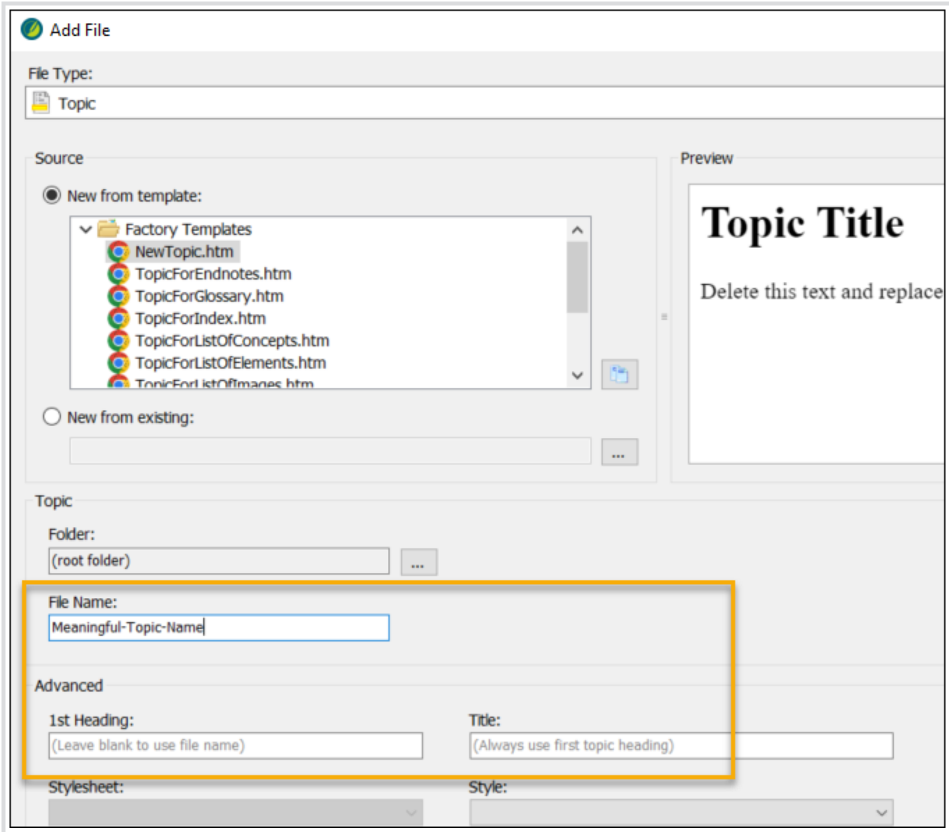
If the topic's file name is Meaningful-Topic-Title, enter **Meaningful Topic Title** in the **1st Heading** field.
If the topic's file name is Herbicide-Tolerant-Plants, enter **Herbicide-Tolerant Plants** in the **1st Heading** field. In this case, the hyphen between "Herbicide" and "Tolerant" is grammatically necessary.

7.3 Blank Topic Titles

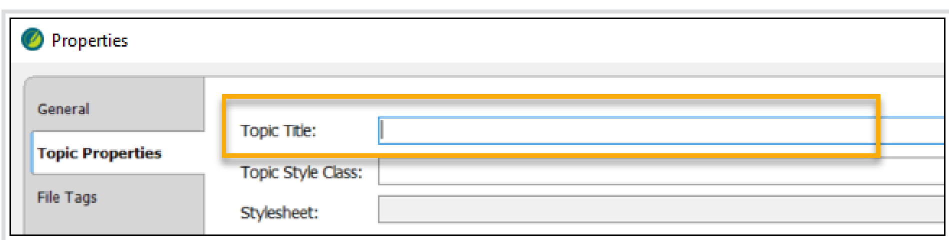
This topic explains why topic titles often appear to be missing in Flare.

☆ A blank topic title field means that Flare is using the first heading of the topic for the title. So the topic isn't actually missing a title. Read on to understand more.

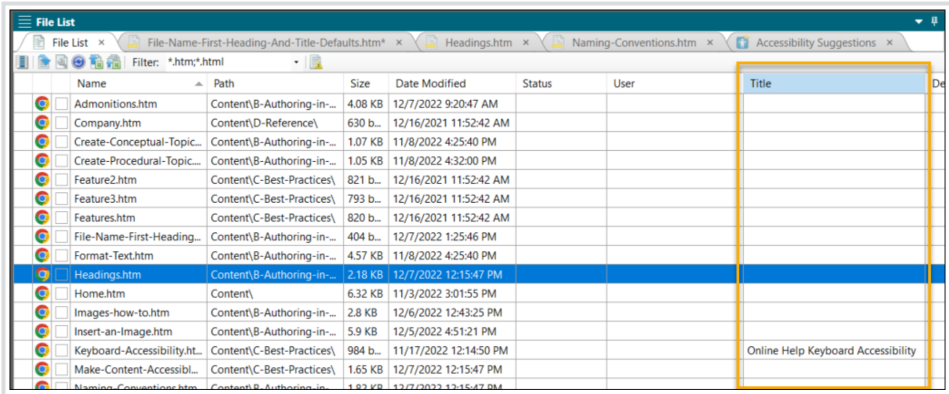
When you create a new topic, you enter a file name for the topic. By default, Flare uses the file name for the first heading in the topic, and for the title of the topic.



When a topic title is inherited from a first heading, the **Topic Title** field in the topic's properties appears to be blank.



In addition, if you view a project's file list (**View** > **File List**), the **Title** column is blank for all topics that inherit their titles from first headings.



In Flare, a blank title field doesn't indicate that a title is missing; rather it shows that the topic's first heading is being used for the title. This is worth noting because of a 508 requirement stipulating that all topics have a title (screen readers read page titles to help users understand the context of a page). In short, Flare doesn't let you overlook this 508 requirement.

7.4 Create Keyboard Shortcuts in Flare

To assign a keyboard shortcut in Flare, you record a macro and then assign a keyboard shortcut to the new macro.

1. Record a macro:
    a. Open a topic in Flare’s XML Editor and place your cursor in a paragraph.
    b. On the Tools tab, click Record.
    c. In the New Macro dialog, enter a name (such as Apply Note).
    d. Using the drop-down list of style on the Home tab, select the required style (for example, p.Note).
    e. On the Tools tab, click Stop.
2. Assign a keyboard shortcut to the new macro:
    a. On the File tab, click Options.
    b. Ensure that the Keyboard Shortcuts tab is selected.
    c. Using the Command Scope dropdown, select XML Editor.

d. Scroll down the list of options to find the macro (prefixed with Playback:).
e. Select a Key Assignment and Modifier in the normal way.

For more information, see Keyboard shortcuts in MadCap Flare.

8 Accessibility

The information in this and the following topics is still being developed.

The topics in this section describe 508 requirements for the documentation that the Metis Tech Docs team creates.

This information covers the elements currently in use in our documentation, and may require updates if we adopt new elements, such as multimedia content.

8.1 Make Content Accessible

Include a meaningful page title. By default, topic titles are set to match the first heading in your topics. Therefore, ensure that the first heading in your topic follows the recommendations in the topic Headings (on page 24).

Ensure that you enter alt text for all of the following elements:

- Cross-references
- Hyperlinks
- Images

Tables

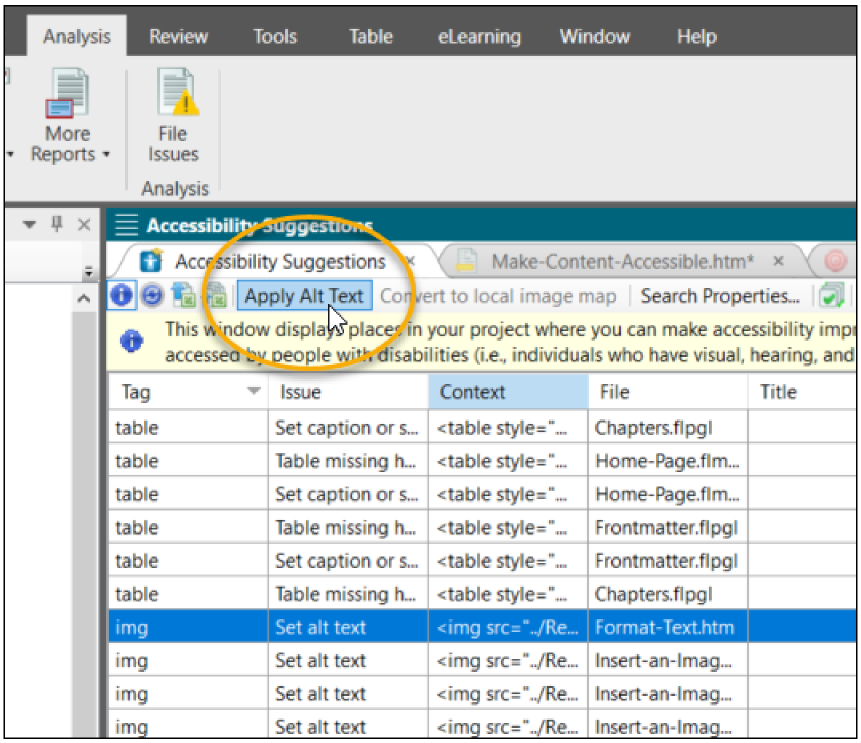
Ensure that you include a header row and headings.

Flare provides an Accessibility Suggestions report to help you find and fix accessibility issues.

8.2 Find and Fix Accessibility Issues

To find accessibility issues:

Select Analysis, then Suggestions, and then Accessibility Suggestions.



9 Reference

The topics in this section describe useful reference information.

9.1 CD Flare Project Folder Structure

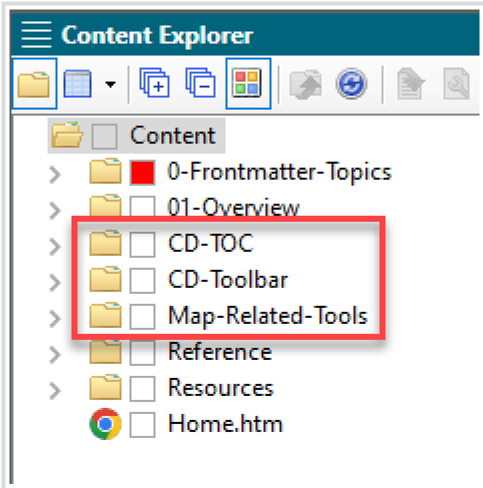
The folder organization of the Flare CD project is mainly based on CD's user interface (UI).

The major areas of the CD UI are the Table of Contents, the CD Toolbar, and the map itself.

Therefore, the Content folder of the CD Flare project includes the following folders:

- CD-TOC
- CD-Toolbar
- Map-Related-Tools

> **Note**: By default, the Content folder is the top-level folder in the Content Explorer.



In addition, the CD Flare project includes the following folders:

- Frontmatter-Topics

  Contains files used in PDF output.

- Overview

  Contains introductory topics.

- Reference

  Shockingly, this folder contains reference topics.

- Resources

  Contains files such as images, snippets, and stylesheets.

CD Toolbar Sub-Folders

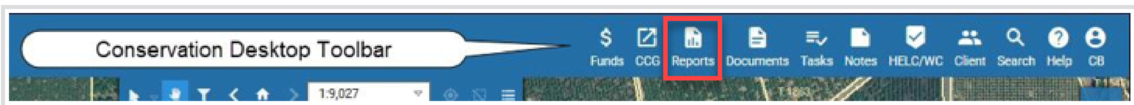
The CD Toolbar comprises the following modules:

- Programs
- Links
- Reports
- Documents
- Tasks

- Notes
- HELC/WC
- Client
- Search
- Help
- User

In the Flare project, the CD Toolbar folder contains sub-folders for each of these modules.

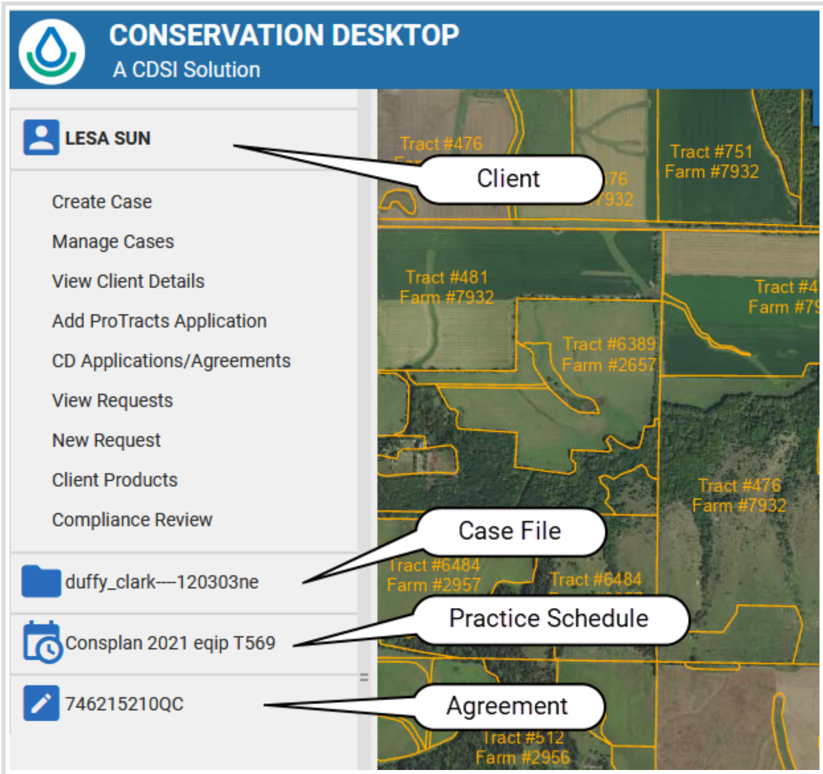
Assistance Requests are a type of report available in the Reports module, which is located in the CD Toolbar.



Therefore, the folder path to Assistance Request topics is Content/CD-Toolbar/Reports/Assist-Requests.

Table of Contents Sub-Folders

The Table of Contents consists of four panels: the Client Panel, the Case File Panel, the Practice Schedule Panel, and the Agreement Panel.



Therefore, the CD-TOC folder contains the following sub-folders:

- Agreement
- Case-File
- Client
- Practice-Schedule

Note that in the Case File panel and in the Practice Schedule panel, there are links that open the PLU Editor and the Practice Editor. Although these editors are launched from these panels, the PLU Editor and the Practice Editor are sub-folders of the Map-Related-Tools folder.

CD Flare Project Miro Map

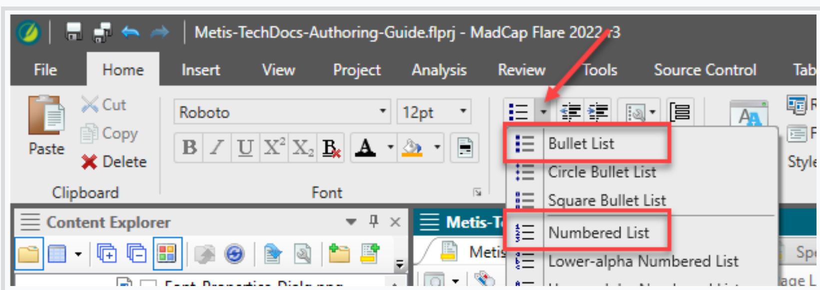
A limitation of Flare is finding files in the Content Explorer. Although the Metis Tech Docs team tried to make the folder structure as intuitive as possible, not every CD topic can be neatly organized by UI location. In addition, anyone unfamiliar with CD may have a hard time understanding the folder structure. Therefore, Metis Tech Docs created a [Miro board] to map out the folder structure of the CD Flare project. You can use Ctrl+F to search for a topic, and then see what folder it's in.

Note the following:

- Because it was time-consuming to create, the Miro map only shows folders for topic files and images. For example, the Resources folder contains sub-folders for stylesheets, page layouts, and other items, but the Miro map doesn't show these sub-folders.
- The Miro map doesn't go further than the h3 heading level. However, this is generally enough to get you in the vicinity of the topic you're looking for.

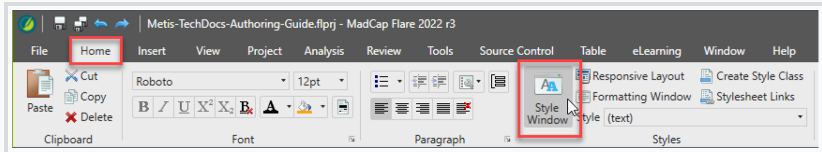
9.2 Flare Styles Cheatsheet

The following table summarizes some of the styles Metis Tech Docs uses frequently, and their locations in the MadCap Flare interface:

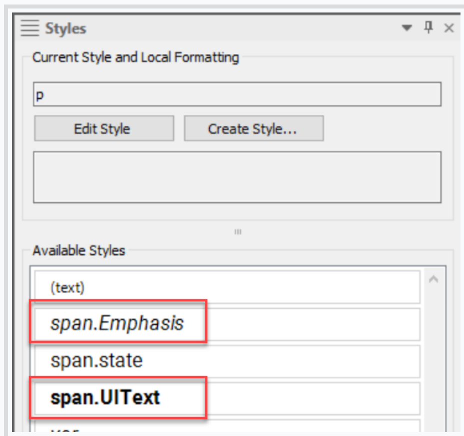
Style	Location in Flare UI
Bullet (Unordered) List Numbered (Ordered) List	

<table>
<tr><th>Style</th><th>Location in Flare UI</th></tr>
<tr><td>Bold
Italic</td><td>

- Highlight the text you want to bold.
- On the **Home** ribbon, select **Style Window**.



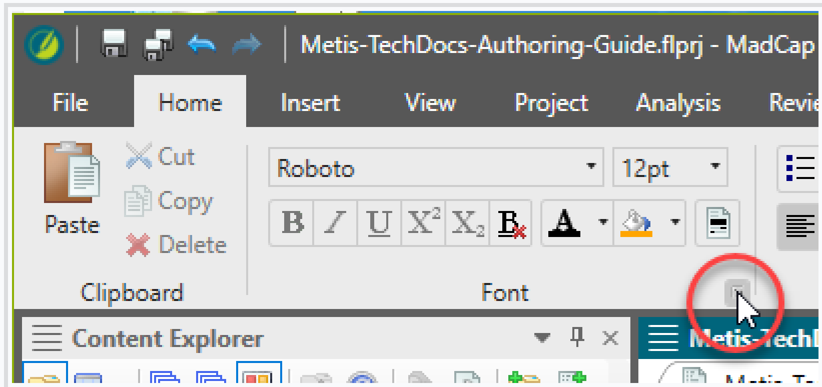
- In the **Styles** panel, select **span.UIText** for bold, or *span.Emphasis* for italic.



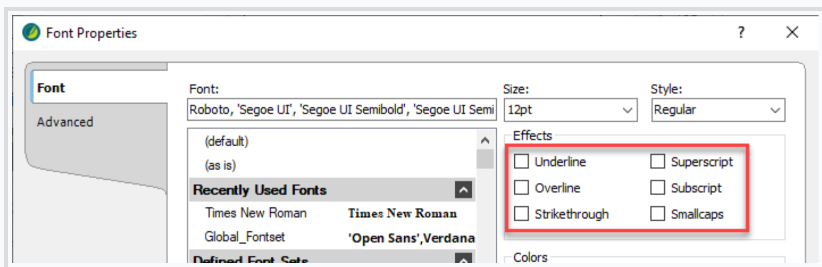
</td></tr>
</table>

<table>
<tr><th>Style</th><th>Location in Flare UI</th></tr>
<tr><td>Underline*
Strikethrough*
Superscript*
Subscript*</td><td>

- Highlight the text you want to style.
- Open the Font Properties dialog.



- Apply a style in the **Effects** area only.



</td></tr>
</table>

*These styles are extremely rare.