Reusing content in DITA: basics

Contents

Introduction to reuse

- Demonstrate the advantages of reusing content
- Analyze content for reusability
- Explain the types of reuse available in DITA

This lesson shows how reuse in DITA can improve the content development process and reduce costs.

Housekeeping and sample files

Download now. It contains the sample files for the entire introduction to reuse course. Extract the contents and put them in a directory that you can access easily.

Inside the samples folder, you will find the following sub-folders:

- exercises
- samples

Each lesson will instruct you on which folders and files to use for the samples and exercises. Save your file as you complete each step to avoid losing your work.

Create a local copy of each file to work in as you complete the lessons. That way, if you reach a point where your working file doesn't match the examples, or is broken for any reason, you can make a fresh copy and resume your work or start over.

In the instructions and examples, we show you the DITA code for each sample file. Most DITA editors have auto-complete or other similar features to guide you through the process of adding elements (for example, if you type the opening tag of an element, most DITA editors will automatically add the closing tag for you). Therefore, you will probably not need to create every piece of code from scratch as you work. Our demo videos were created in *oXygen XML Editor* and show the differences between working in author view, which presents the DITA content in a user-friendly visual format, and working in text view, which shows the DITA code.

Reasons for reuse

Reuse—storing content in a single source and using it in multiple places where relevant—is one of the benefits of maintaining content in a structured form such as DITA. Reusing content offers the following advantages:

- Reduces replication. Reusing content eliminates the need to copy and paste information, or to write nearly identical content multiple times in multiple locations. This saves time and helps ensure greater accuracy.
- Reduces costs. By reusing content, you reduce the cost of time that would have been spent duplicating information manually. This makes content creation faster and eliminates multiple reviews. Reuse reduces localization costs. Replicated content is translated separately every time it appears, but a single source of content that is reused is only translated once.
- Increases consistency. Every time you manually duplicate content, you introduce the risk of human error. Having multiple copies of the same information also increases the risk that the content will be updated in some places but not in others, or that the various copies will become slightly different from each other over time. Maintaining a single source means that the reused content will always be up-to-date and consistent wherever it is used.
- Allows product- or customer-specific output. You can combine different reusable topics in different maps for different products or customers. For example, you might have a series of product manuals that contain some information that's identical and other information that's more product-specific. Rather than writing a separate manual for each product, you can combine product-specific topics with common topics in a map for each product, which allows you to write each of the common topics only once.

Implementing reuse is a long-term investment. You will usually see the benefits of reuse in update or maintenance costs. Most of these benefits will occur after, not during, the initial release.

Analyzing content for reuse

Most content contains some information that can be reused. An analysis of your content can help you find what is reusable. You might find entire topics that are reusable, or you might find individual elements (such as paragraphs, notes, or list items) that are reusable.

When analyzing your content you can find:

- No match. If chunks of content do not match each other, you don’t have a case for reuse.
- Exact match. If you find chunks of content (or even entire topics) that are identical, you should reuse them.
- Gray area (inexact or partial match). If you find content that almost matches or is similar to other content, you can probably benefit from reusing it, with some minor changes. Here are some examples of content that fall into this gray area:
  - Naming: Content can be almost the same, except for the occurrence of product or device names, model numbers, or company names.
  - Different processes, definitions, or details: You might have several product manuals that are mostly the same except for certain pieces of information—for example, a different set of installation instructions for one version of a product, or some additional safety information for another.
  - Locale: Different units might be used in versions of content intended for different locales.
  - Different order: The same content might appear in a different sequence from one deliverable to another.
  - Subset of content: If you have a shorter version of your content—for example, a quick guide for a user manual—common content between the shorter and longer versions should be reused.
  - Necessary specifics: Your content might be identical except for a few pieces of information that need to be specific (such as the size or number of screws).

  In some cases, it makes more sense to rewrite content than to try to reuse it as-is:

  - Superfluous specifics: If you have two pieces of content that are almost identical, except for unnecessary specifics—such as the color of a panel or the number of optional components—it is best to rewrite the content using generic terms (or eliminate the specifics altogether).
  - Usage and phrasing: If two pieces of content convey basically the same information, but are written differently,consider merging them into a single reusable chunk (or choose the version that works best in all locations).

Types of reuse

DITA offers four reuse facilities: topics and maps, fragments, variables, and filtering.

Topics and maps: Topic- and map-level reuse is one of the most straightforward ways to reuse content. You can achieve this type of reuse by referencing the same topic in more than one map. Such topics might include boilerplate safety information or common installation instructions that apply to multiple products. Similarly, you can reuse one map in multiple places. For example, the map for a product datasheet might be published individually, and also as part of a larger bookmap for that product's user manual.

Fragments (conrefs): Fragment-level reuse involves pulling shared pieces of content (usually elements) from a common source into a topic. Examples of the shared content include paragraphs, lists or list items, tables, or sections. In DITA, the content reference (or conref) mechanism implements fragment-level reuse. Fragment-level reuse can also be used in DITA maps.

Variables (keys): In variable-level reuse, you add a placeholder to your content that is replaced by a piece of text when you generate your output. The replacement text varies depending on the circumstances. Variable-level reuse is used for inline content, such as company names, product names, URLs, or filenames.

Filtering (conditions): Filtering that allows you to selectively remove certain pieces of content, based on different conditions, to create product- or customer-specific deliverables. For example, when documenting two similar products, a basic version of the product does not have features that are available in a more advanced version. Filtering allows you to create a single piece of content containing all features, then filter out the advanced features when using the content for the basic version.

The first two of these types of reuse—topic- and map-level reuse and fragment-level reuse—are covered in this course. Variables and filtering will be addressed more thoroughly in future courses.

Writing for reuse

When writing for reuse your content must be:

- Consistent
- Context-free
- Generalized

By separating specific information from common information wherever possible, you can make sure that your common chunks of content are generic enough to be reused wherever needed.

To make your content consistent:

- Keep topics short and granular; this maximizes their reuse potential. A short topic with only one heading level can be reused in more places than a long topic that covers large amounts of information and contains multiple sections.
- Establish a style guide for tagging and usage (or to agree on usage if you don't have the time or budget to create a formal style guide).
- Using a neutral voice and a limited vocabulary can ensure consistency and make your topics more reusable. Limit your use of pronouns (particularly gender-specific ones) and avoid idioms and colloquialisms. A neutral, consistent voice will not only help with reuse, but also with localization.

To make your content context-free:

- Do not assume anything about information that comes before or after the topic. Avoid using words such as “previous,” “next,” “earlier,” and “later.”
- Do not assume the type of the document in which the content is used. Avoid using phrases such as “in this chapter” or “in this section.”
- Do not use inline cross-references. In your version of the content, the cross-reference target exists; but it might not be in the next map where your content is reused. Instead, use DITA relationship tables to automatically generate links to releated topics.

To make your writing more generalized:

- Avoid using keywords or product dependencies (wherever possible). For example, if the same set of instructions can be used with multiple products, eliminate product-specific references.
- Don't be too specific unless it's crucial. Avoid including gratuitous modifiers, such as number, size, or color, in your content.
-

To strike a balance between making content generic for reuse and including necessary specifics, consider a table at the beginning of each deliverable that identifies certain product information. By providing that information up-front, you can then remove keywords from the content and rely on generic terms. You can also pull in additional reusable fragments that are free of direct content references.

As with making your content consisten, generic content will not only help facilitate reuse, but will also better prepare you for localization. Reusing content also means that you will have fewer topics overall to translate, which will help reduce localization costs.

Understanding the problem

When first learning about conrefs in DITA, it might be tempting to use it in any number of situations. However, it’s wise to proceed with some caution, especially if your content will be translated. And even if your content is not being translated today, assuming that it will be translated someday will avoid problems when “someday” comes.

In general, just because it works in DITA, and it works well in English, it does not mean that it’s a good thing to do to content that will be translated into other languages. Languages vary in their vocabulary, but it goes far beyond that. They also vary in how words can be put together and in the effect that words might have on the surrounding words.

Not every language works the same as English. Some languages require a word to take a different form depending on whether it's in the subject or object of the sentence. Some languages have grammatical gender, and so words with gender can affect words that are related to it in the sentence. Some languages have a different number of plural forms than in English, where we're used to the concepts of zero, one, and more than one.

Each language has different rules of grammar and syntax. It's not the point of this lesson to understand the rules for every language. Instead, we'll use English to illustrate why substituting words isn't a good idea when your content will be translated into other languages with other rules.

Example:

Casey ate an orange.

What if we were to substitute another word for orange?

Casey ate an apple.

Still OK. But what if we substituted a different word?

Casey ate an tomato.

Tip: "Knowledge is knowing that a tomato is a fruit. Wisdom is not putting it in a fruit salad" - Miles Kington

Now we have a problem. In English, we need the article **a** instead of **an** when the word starts with a consonant sound. Maybe not every language has a rule like that, but English does. Substituting a fruit breaks the a/an rule in some cases.

But that's not the only way we can break our sample sentence.

Example:

Casey ate an apple.

What if we substituted the plural?

Casey ate an apples.

You're probably getting the idea. English has rules that are broken easily by substituting words within sentences. Other languages have other rules that are also broken easily by substituting words within sentences. Just because you're following the rules for English doesn't mean you're not breaking the rules for another language. You probably don't know all the rules for all the languages your English content will be translated into. So, **you should not use conref within sentences (using ph or similar elements) to substitute parts of the sentence.** The same goes for other reuse techniques like conkeyref and conditional filtering (profiling with ditaval). If you use these on parts of a sentence, you're just asking for trouble because you will have bad content when it's translated.

Reusing paragraphs

Whole, *grammatically independent* paragraphs can be reused safely *within the same context*. This would include the p element, or any element that grammatically is a paragraph, even if it has only one sentence. Let's break that down.

Grammatically independent

A paragraph that is grammatically independent doesn't rely on anything outside the paragraph to complete its meaning. For example, if pronouns within the paragraph refer to something outside the paragraph then it would not be grammatically independent. Substituting pronouns for nouns to make a paragraph less specific does not make it a good candidate for reuse. Substituting less specific nouns, however, is a good strategy. "Your computer" in place of "The Compu-Master 5000", for example.

Example:

Yes: Your computer is equipped with an attractive protective case. To avoid losing your computer, please do not paint or wallpaper the case to match your décor.

No: This may not appear because it requires an additional license. Please call our office for more information.

("This" and "it" refer to something outside the paragraph.)

Within the same context

Some words have multiple meanings that still make sense in English even if they're not being used in their original context. Take the word "key" as an example. It might refer to a small piece of cut metal that's used to open a lock, or it might refer to something on a rotating shaft that keeps another machine component such as a gear rotating along with the shaft. The word "key" also has other meanings related to maps or cryptography.

If we were told to "examine the key for defects", it might make sense to us in English in all of those contexts. In our minds we substitute a different concept for the word "key" depending on the context. But we can't count on those different concepts sharing the same word in other languages.

This can be a tricky one to spot, because we usually substitute the correct concepts in our minds without even being conscious of it. So here's another example. In English the word "battery" could refer to something in an automobile or to something in a watch. Both supply electrical current, so we might think that it's the same concept and context. But when we stop and think about it, a watch is not at all the same context as an automobile and their batteries are very different in their construction. So it's not safe to reuse between those two contexts, even if there are similarities in the concepts. And in fact, different words are used for those two kinds of batteries in some languages.

The main point to keep in mind is that in other languages an entirely different word might be used for different concepts or contexts, so reusing in English across different contexts is not helpful for translated content.

Reusing steps

Like paragraphs, whole grammatically independent steps can be reused in the same context.

Yes: "Close the window."

No: "Close it."

Structured parts of steps

Step Example, Info, Choices, Tutorial Information, and elements like them can be reused if they are grammatically independent and in the same context.

Reusing list items

You can probably already guess that a list item that you want to reuse needs to be in the same context and grammatically independent - without pronouns that refer to something outside the list item. But there's more to it because lists are often introduced by an incomplete or complete sentence. The incomplete introductions are sometimes called stem sentences. A list preamble is a broader term that would include the complete sentence introductions.

Items in ordered or unordered lists that are independent clauses or are introduced by independent clauses can be reused and conditionally filtered. "What's an independent clause?", you say? It stands on its own and expresses a complete thought.

The problem with the incomplete sentences introducing a list is that in some languages the list items that complete the sentence need to be inflected - changed - based on the stem that introduced them. If you were to reuse the items with a different stem, it might not be correct. If the list preamble relies on the list items to complete the sentence, you should rewrite the preamble so that it stands alone grammatically.

Yes: You can choose one of the following colors:

- yellow
- red

- green

No: You can ride:

- a bus
- a bicycle
- a train

Storing your reused content

It’s good practice to hold your reusable elements in warehouse topics. These are topics that are written specifically for reuse rather than for direct publication. By following this practice you will avoid creating “spaghetti references” - referenced elements at random points within a publication. Having unpredictable referenced element locations means that you can’t easily manage the dependencies of one topic to another.

By keeping your reusable elements in defined locations, your referenced element won’t be removed from the map unintentionally. As long as the warehouse topics are included in the publication map, their referenced elements will be available to resolve conrefs.

Some component content management systems may have specific requirements for where reused content must be stored. For example, some systems require that all content included in a map, including content included solely for reuse, be stored in a folder beneath the folder that holds the map. Check the requirements of your CCMS if you’re in doubt.

Reusable maps and topics

- Demonstrate reuse at the map and topic levels
- Create reusable topics
- Write content in ways that maximize its potential for reuse

This lesson shows how to create reusable maps and topics in DITA, and how to write content with reuse in mind.

Reusable maps and topics

In the simplest form of reuse, you can reuse a DITA topic in any number of maps.

You can reuse a topic in multiple maps

You can reuse a map in multiple ways (for example, publishing it as a standalone deliverable or including it in other maps

Be careful about who edits your reusable topics or maps. Make sure that no one adds phrases such as "in the previous chapter" to your topics, or writes content that depends on the other topics in one of the maps where the reused topic is referenced. Remember that a change in a reused topic will be reflected everywhere that topic is referenced. The same applies to reused maps.

Creating a reusable topic

Assessment

Answer the following questions to check how well you understand the concepts in this lesson.

Across publications

The following is a good candidate for reuse across publications about automobiles: “In case of an emergency, move your vehicle to the side of the road, if possible.”

- True
- False

The content would be reused in the same context, and does not contain pronouns that refer to something outside itself. This content is a good candidate for reuse.

Candidate

The following is a good candidate for reuse: “Don’t overtighten it or damage could occur.”

- True
- False

The pronoun it could refer to anything, and that might affect how the sentence would be translated. It would be better to substitute a generic noun in place of “it”, such as “the fastener”.

Stored

Where should reused content be stored?

- You can reuse content wherever it might be, just as long as it's referred to in one of the maps in the publication where it's being reused.
- Topics that contain reusable elements should be dedicated to that purpose and should only be used as a resource rather than being published directly.

Even though it might be technically possible to do this, it's not a good idea. Using dedicated "warehouse topics" makes it easier to manage the reuse.

Translation

Using phrase elements to substitute words within sentences never causes problems for translation.

- True
- False

Not all languages follow the same rules as English. Just because the substitution works in English doesn't mean that it will work well in other languages. And, you should assume that your content will be translated someday, even if it's not translated today.

Using conrefs

- Define the limits on element reuse
- Demonstrate element reuse with conrefs
- Explain element requirements for conrefs

This lesson shows how to create and use conrefs to achieve fragment-level reuse.

What is a content reference?

A content reference (conref) enables you to reuse elements from one topic in others.

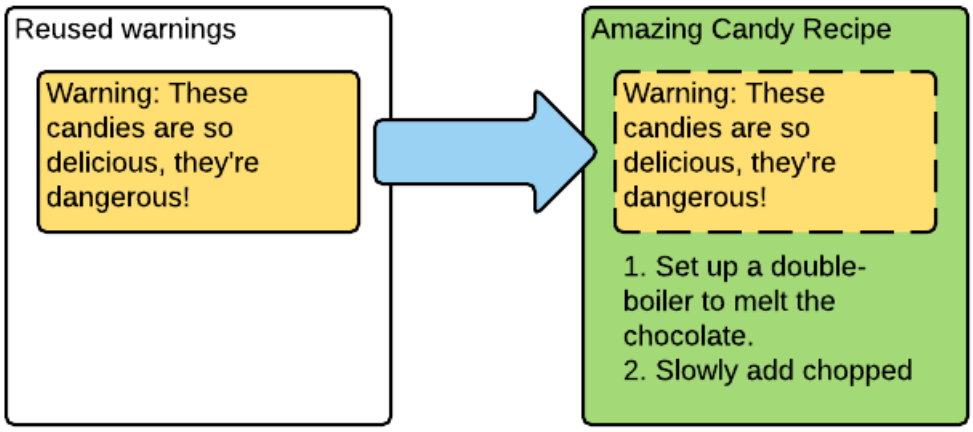
When pieces of content are repeated in multiple topics or publications, using conrefs can make your content easier to maintain. Unlike copy and paste, where content is duplicated, conrefs insert reused elements *by reference* into a topic.

Note:

For more information about conrefs, see the *OASIS DITA Version 1.2 Standard*.

Conref basics

The original reused element stays where it is, and a content reference (conref) to that element is inserted in each topic where the reused element and its content should appear. If the original content changes, those changes also appear where that content has been referenced, once the references are resolved by a DITA authoring or publishing tool.



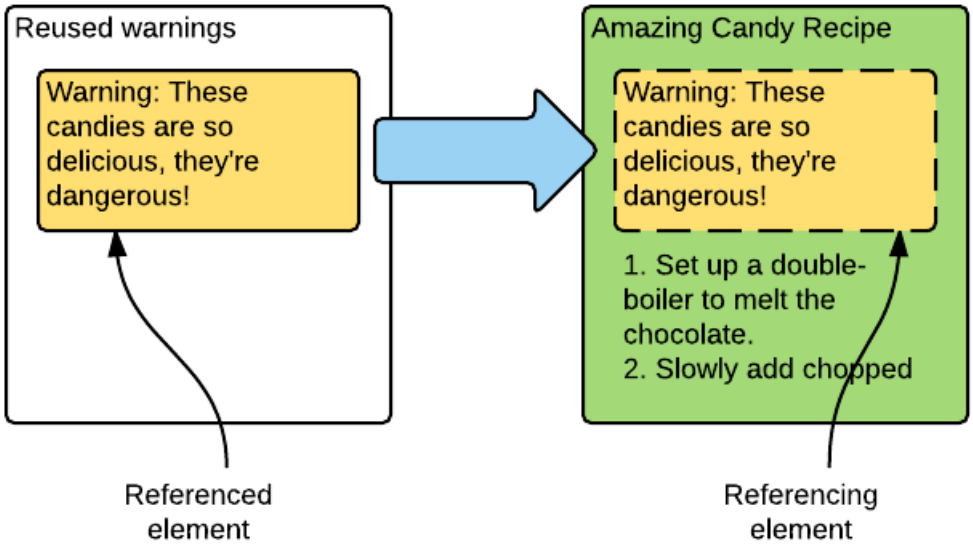
A conref has two parts:

referenced element

the element that contains the content you want to reuse

referencing element

the element that marks where the reused content should be inserted



Tip: In writing about content references, the word conref is often used as a verb to mean "reuse an element by making reference to it using the conref attribute."

Practice

Conrefs in action

When you conref an element, the element itself, and all its contents are inserted at the location of the referencing element. This includes any other elements that the referenced element contains. This means, for example, that if a referenced paragraph contains a <uicontrol> element, the paragraph and the <uicontrol> element contained within it are inserted at the location of the referencing element. This also means that a reference to a <ol> element includes all the <li> elelments that are contained by the <ol> element.

A conref directly references a reused element using its location and ID. The location is the path to the topic that contains the reused content, and the id is the ID of the element to be reused. In DITA not every element requires an ID, but it is required for any reused element.

Note: If the element you want to reuse does not have an ID your content management system might prompt you to create an ID for the reused element as one of the steps in creating the conref. An ID must start with a letter, number, or underscore.

Practice

The conref attribute

Instruction

The conref attribute holds a value that identifies the referenced element. In the example below, the warehouse topic warnings.dita contains a reusable note.

```
<topic id="warehouseWarnings">
...
<note type="danger" id="hotWarning">Surfaces are hot.</note>
```

```
...
```

Figure 1: The warehouse topic containing a reusable note

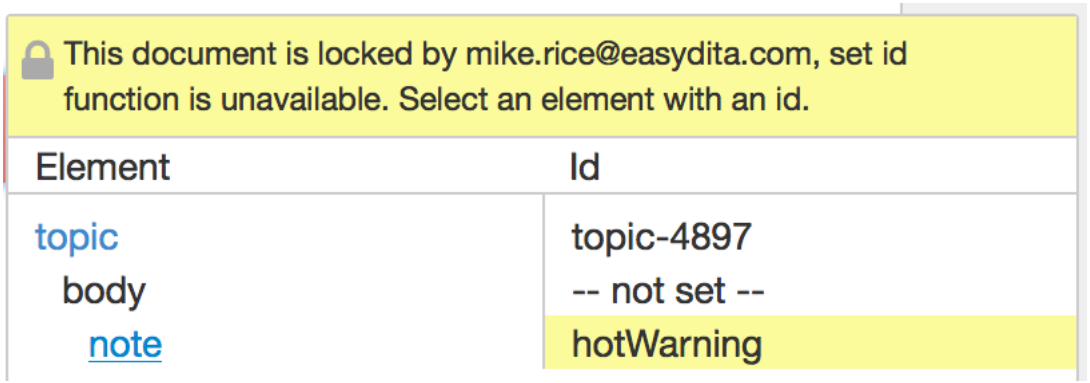
In this example, the files are stored in a local filesystem, and the warehouse topic with the note you want to reuse is stored in the warehouse folder. You can reuse the note in another topic by making reference to the note's id using a value in the conref attribute of an empty note element:

```
...
<note conref="../warehouse/warnings.dita#warehouseWarnings/hotWarning"/>
...
```

Figure 2: Using the reusable note in a topic

Note: A reused element must have an id. ids must start with a letter, number, or underscore.

Your DITA authoring tool probably will help you fill in the conref attribute value to correctly identify the location of the reused target element as in this easyDITA example:



The following table shows some examples of how the conref attribute value is formatted in various situations. You would replace the placeholders in these examples:

- **topicID** is the id of the topic that contains the reused content
- **targetID** is the id of the reused element
- **folder** is a directory in a local file system
- **file.dita** is the name of a DITA file
- **http://example.com** is the name of a network computer

Target location	Conref attribute value
In the same topic	#topicID/targetID
In a topic in a local file system	folder/file.dita#topicID/targetID
In a topic at a network location	http://example.com/file.dita#topicID/targetID

Practice

Element requirements for conrefs

Instruction

In most cases, you will conref only elements that are alike, so that both the referenced element and the referencing element are the same type of element. There are few exceptions, and the rules that allow these exceptions are complex and beyond the scope of this course.

Some elements are invalid without other elements inside them. For example, in a task, the <step> element requires a <cmd> element within it. When elements have required contents, those elements must be present in the referencing element and in the referenced element. This requirement can be satisfied by inserting empty required elements inside the referencing element. These empty elements are replaced with the actual required elements when the conref is resolved.

In the following example, the empty <cmd> element is a necessary part of the referencing element for the step.

```
<task id="referencing-element">
  <title>A task that reuses a step</title>
...
    <step conref="../warehouse/task-reuse.dita#warehouse-task/reusedStep">
                <cmd/>
            </step>
...
</task>
```

Figure 3: The referencing <step> element with an empty <cmd> element

```
<task id="warehouse-task">
  <title>A warehouse task topic</title>
...
    <step id="reusedStep">
                <cmd>Lock out and tag the power source.</cmd>
            </step>
...
</task>
```

Figure 4: The referenced <step> element in a warehouse topic

Practice

Reusing across topics

Instruction

When publishing a topic that reuses content from another topic, the content references can be resolved only if the referenced elements are reachable through the publication map. For example:

- If the topics containing the referenced element and referencing element are in the same map then conrefs between those topics can be resolved through the context of that map.
- If the topics containing the referenced element and referencing element are in separate maps but those maps are part of the overall map structure of the same published map, the conrefs between those topics can be resolved through the context of the publication map.

In a later lesson you'll learn about best practices for reusing content. Among these practices is using warehouse topics - storing your reused content separately from the main body of content.

Ordinarily, if a topic is included by a topicref in one of the maps, it will be published along with all the other topics. But clearly you wouldn't want to publish a warehouse of reusable content along with your publication. To mark resources of this sort that aren't intended for publication, set the topicref's processing-role attribute to resource-only.

In the following example, task.dita will be published, but warnings.dita will not. However, warnings.dita is available for resolving references from task.dita or any other topics in the map.

```
<map>
...
  <topicref href="warehouse/warnings.dita" processing-role="resource-only"/>
...
  <topicref href="topics/task.dita"/>
...
```

```
</map>
```

Figure 5: Using resource-only in the topicref processing-role attribute

Note: When working in a component content management system, the system might resolve conrefs and other direct references that appear in a topic you're viewing or editing, even without a common context set through a map. But, depending on your publishing configuration, those references might not resolve when the topic is published. The best practice is to have topics with the referenced element and referencing element in the same map or one of its submaps.

Practice

Attributes of content references

Instruction

Sometimes your reused content will use attributes. For example, a <tm> element is required to have a tmtype attribute. Or, you might conref an element that has conditional filtering values set. When you reuse an element that has an attribute value, which value gets used in the referencing element? Is it the value from the referenced element or the value from the referencing element?

The answer is that by default, the referencing element value gets used if there is one. If the referencing element doesn't have a value, then the referenced element value gets used, if there is one. The exception is when the referenced element uses the value -dita-use-conref-target. The referenced element value will be used in this case. For elements where an attribute value from a defined list is required, the -dita-use-conref-target value enables the referenced element value to be used, just as it would if the referencing element value were blank.

Table 1: Results of merging the values of referencing element and referenced element values in conrefs

		Referencing Value			
Referenced Value		**blank**	**not blank**	**-dita-use-conref-target**	**xml:lang**
	blank	blank	referencing element value	blank	referencing element value
	not blank	referenced element value	referencing element value	referenced element value	n/a
	-dita-use-conref-target	referenced element value (of the referenced element's referenced element)	referencing element value	referenced element value (of the referenced element's referenced element)	n/a
	xml:lang	referenced element value	n/a	n/a	referenced element value

Note: Normally, you wouldn't use -dita-use-conref-target as a referenced element value, since it implies that the element with that value is the referencing element of a conref. But it could be the case that you conref an element that itself is a conref. In this case, if both elements with a conref attribute have the same attribute set to "-dita-use-conref-target", the value of the attribute will match the value in the referenced element of the referenced element.

Practice

Best practices for using conrefs

What might seem to work well when using conrefs in English content might not work well when the content is translated. Here are a few guidelines that will make more sense after completing this lesson:

- Don’t use conref with ph to replace parts of a sentence.
- Use extreme caution when using conref to reuse whole sentences.
- When you do conref, do it with whole block elements that meet the following requirements:
  - The reused content is grammatically independent.
  - The referenced element and referencing element are within the same context.

Practice

Exercise

Assessment

Answer the following questions to check how well you understand the concepts in this lesson.

Referencing element

For the following conref referencing element sample, select all the answers that are true.

```
...
    <step conref="../warehouse/task-reuse.dita#warehouse-
task/reusedStep">
                <cmd/>
            </step>
...
```

- The <step> element is missing required content.
- The <cmd> element is missing required content.
- The <cmd> element is not required here because the <step> element is a referencing element and its contents will be replaced when the conref is resolved.
- The <cmd> element is required here even though the <step> element is a referencing element.

The content must be valid both before and after the conref is resolved. Since the <step> element requires the <cmd> element, the <cmd> element is required even though it will be repaced when the conref is resolved. The <cmd> element does not require any content. So, this example has all of the required structure to be valid before the conref is resolved.

Resolve

A conref will always resolve correctly when published even if the topic that contains the referenced element is not in the published map nor any of its submaps.

- True
- False

Sorry. If your content is in a content management system, it might appear that any conref will resolve correctly regardless of whether the referenced element and the referencing element are reachable through the same map. But, your publishing configuration might require that the topic containing a referenced element for a conref be

referenced in the published map or one of its submaps. The best practice is to reference all of the required content in the published map.

Topic

If you need to include a topic in a map for reuse purposes only, how should you do it? Choose the best answer.

- Use a topicref to include the topic in a submap with the toc attribute set to “no” on the topicref to the submap.
- Use a topicref to include the topic in the map, and set the proceesing-role attribute of the topicref to “resource-only”.
- Use a topicref to include the topic in the map, and set the toc, linking, and search attributes of the topicref to “no”.

Setting three attributes - toc, linking, and search - to “no” has the same effect as setting the processing-role attribute to “resource-only”, but setting just one attribute is a simpler solution.