# Writing Effective DITA Task Topics

## Michelle Carey

## Shannon Rouiller

# Writing task topics

## Write one task topic for one task.

### Don’t put multiple tasks in one topic. Remember that we might want to reuse that task topic in another context.

### The more self-contained the topic is, the easier it is to reuse.

## Tell users how to accomplish a goal.

## Provide procedural details in numbered steps.

#### Some tasks will have only one step. Such tasks are fine.

## Sometimes, use choices rather than steps.

## Provide context (where, when, or why to perform task), examples, and links to prerequisite and postrequisite tasks.

## Focus on the user goal, not the product function.

## Link to the next task (if any), concepts, reference, or other related information. Don’t strand the user.

## break

## Include an opening paragraph with the <shortdesc> element to briefly introduce the task.

## If the task has prerequisite tasks, describe or provide links to the prerequisite tasks in the <prereq> element before the step list.

## Use the <steps> element for ordered steps, not <ol>.

## Use <steps><step><cmd> even for a one-step task.

## Write one step for each significant user action but combine simple steps with their significant steps. For example, watch for single steps that say “Click OK.”

## Write the <cmd> parts of the steps as brief imperative sentences. When the user's context has changed, introduce the step with a context phrase, for example, “On the xyz page, . . . .”

## break

## If a task has more than nine steps, consider removing steps, combining steps for logic and flow, or dividing the task into two or more separate task topics.

## Do not go more than 2 levels of depth in steps.

## Describe only one way of doing a task in each topic.

## Tag optional steps correctly: set the importance attribute on the <step> tag to optional.

# Tagging

# Task topic elements (task type)

## Shortdesc

## Context

## Stepsection (new in 2010)

## Step and cmd

## Info

## Stepresult

## Stepxmp

## Substep

## Choices

## Choicetable

## Example

## Postreq

## result

# Task topic elements (domains)

## Uicontrol

## Userinput

## Codeblock

## Codeph

## Menucascade

## Wintitle

## cmdname

# Linking, DITA maps, rel tables

## DITA map links: Do most of your linking through the DITA map:

### If the task is one of a set sequence of tasks, provide a link to the next task. The parent link is generated automatically if you set the container topicref to collection type = sequence.

### If you have just one task, you can use the automatic linking from the DITA map to create links to parent task topics and to closely related concept, task, and reference topics. type=collection or don’t set the type.

## Rel table links: Use for related links

## Inline links <xref>: Often used in the <prereq> tag.