

## Rewriting short descriptions

Sometimes it's difficult to decide whether a short description should use natural language. This topic provides some guidance.

Consider the short descriptions for the elements in the programming domain. When this domain went to review in January 2023, the short descriptions did not implement natural language. The following table contains commentary:

Element	Original short description	Comments	Revised short description
<apiname>	The <apiname> element identifies the name of an application programming interface (API), such as a Java class name or method name.	Poor candidate for natural language. A natural-language short description will tend to seem circular.	
<codeblock>	The <codeblock> element identifies lines of program code.	Do not use natural language. A <i>code block</i> has a <a href="#">specific meaning</a> in programming that does not match how we use it in DITA.	
<codeph>	The <codeph> element identifies a code snippet.	The current short description is problematic, as the term <i>code snippet</i> has a <a href="#">specific meaning</a> in programming that does not match how we use the term in DITA.	A code phrase is a small portion of source code, machine code, or text that is displayed in-line.
<coderef>	The <coderef> element references an external file that contains literal code.	This is iffy and needs to be approached carefully. The short description for <topicref> should be the model for how to recast this in natural language.	A code reference is the mechanism for referencing an external file that contains literal code.
<option>	The <option> element describes an option that can modify a command or a configuration.	This is iffy. The term <i>option</i> is too broad to use as a natural-language replacement, and <i>command or configuration option</i> seems cumbersome. Perhaps <i>command-line option</i> ? However, that might exclude ways in which current DITA users have used <option> to describe GUI parameters.	[?] A command-line option is a command used to pass a parameter to a program. These entries, also called command-line switches, can pass along cues for changing various settings or running commands in an interface.
<parname>	The <parname> element identifies the name of a parameter.	Poor candidate for natural language. A natural-language short description will tend to seem circular.	
<parml>	The <parml> element identifies a specialized definition list that is designed for documenting parameters.	Good candidate for natural language. It easily can move to the format "An item is a ..."	A parameter list is a specialized definition list that is designed for documenting parameters.

Element	Original short description	Comments	Revised short description
<plentry>	The <plentry> element contains one or more parameter terms and definitions.	Good candidate for natural language. It is a container element, and it easily can move to the format "An item contains ..."	A parameter-list entry contains one or more parameter terms and definitions.
<pd>	The <pd> element specifies a parameter definition within a parameter list entry.	This is iffy. We can model it after how we defined <dd>, but does that really add value?	[?] A parameter definition is a definition of a term that is defined in a parameter-list entry.
<pt>	The <pt> element specifies a parameter term within a parameter list entry.	This is iffy. We can model it after how we defined <dt>, but does that really add value?	[?] A parameter term is a term that is defined in a parameter-list entry.