Sandcastle Help File Builder (SHFB)

A Reference to Sandcastle's XML Documentation Comments

Michael Sorens ● Copyright © 2010-2011 ● V1.0.4 ● 2011.08.03

This quick reference accompanies **Taming Sandcastle: A .NET Programmer's Guide to Documenting Your Code** at http://www.simple-talk.com/dotnet/.net-tools/taming-sandcastle-a-.net-programmers-guide-to-documenting-your-code/.

Anatomy of an Entry

tag • compiler validated • platforms • required • choice of values • optional

`<sample>` ✔ .NET SHFB S root ● mixed   p=*value*   q="abc"|"5"   [n=*int*]

Sandcastle • place holder • literals

`<c>` .NET SHFB S child/inline ● text
Format short text content as code.
- Formats code fragments inline; compare to <code>.

`<code>` .NET SHFB S child/block ● text|element
- **Language** [lang=*language_ID* | language= *language_ID*]
- **Text** [region=*region_name*] [source=*file_name*] [title=*text*]
- **Numerical** [tabsize=*number_of_spaces*]
- **Boolean** [filter] [keepSeeTags] [numberLines] [outlining] [removeRegionMarkers]

Format block as code with syntax highlighting for supported languages (default C#).
- Child: <code>
- SHFB adds all above custom attributes to the .NET standard <code> tag.
- Only source and region attributes may be used when nesting <code> elements.
- If title is omitted, language becomes the title; to suppress, set title to single space.
- Attributes are supported by SHFB only, not .NET or Sandcastle.
- Filter attribute applies only to VS2005.

SHFB Help ► Conceptual and Additional Content ► Additional Content ► Importing and Colorizing Code From Source Files
SHFB Help ► Custom Build Components ► Code Block Component

`<example>` .NET SHFB S root ● mixed
Generates an eponymous output section.
- Typically includes <para> and <code> elements.

`<exception>` ✔ .NET SHFB S root ● mixed   cref=*member*
Specifies exceptions that can be thrown from the current member.
- Compiler does not require <exception> tags but will validate cref value if used.

`<exclude>` SHFB root ● empty
Suppresses documenting the current member in the generated output.
- Supported by SHFB but preferred approach is to use the ApiFilter project property.

`<filterpriority>` root ● text
Used by the VB editor to filter elements in the auto-complete popup.
- Not used by Sandcastle at all. Valid values are 0 to 3. See Scott Dorman's blog.

`<include>` ✔ .NET SHFB S root ● empty   file=*file_name*   path=*XPath_expr*
Imports a block of XML comments from an external file.
- Select all or part of an XML file with an XPath expression.

`<inheritdoc>` SHFB S root|child/block ● empty   [cref=*member*] [select=*XPath_expr*]
Inherit common information from ancestors.
- Must use cref attribute unless at root level.
- <overloads> tag is never inherited; use select="overloads/*" if needed.
- <param> tags are only inherited if names match base class names.
- Use cref and/or select, plus nested tags, to refine inheritance.

SHFB Help ► Getting Started ► Using the <inheritdoc /> Tag

`<list>` .NET SHFB S child/block ● element   type="bullet"|"number"|"table"|"definition"
Create a list or table.
- Child: <listheader> (first child if present) and <item> (multiple instances allowed), each of which contain successive children <term> and <description>.
- Table, bulleted list, or numbered list require just <description> in each <item>.
- Definition list (not officially supported) requires both <term> and <description>.
- Table requires only a <term> in the <listheader>, if present.

`<note>` SHFB S child/inline ● mixed   type="caution"|"inheritinfo"|"implementnotes"
Format text block as a note.
- Use this for a paragraph that should stand out.

`<overloads>` SHFB S root ● mixed
Adds documentation specific to the overloads page of an overloaded member.
- May contain any root elements.

`<para>` .NET SHFB S child/block ● mixed
Structure a text block as a paragraph.
- Use to structure your text containing multiple paragraphs.

`<param>` ✔ .NET SHFB S root ● mixed   name=*param_name*
Describes an ordinary (non-type) parameter of a method or constructor.
- Should be a sentence—not just a fragment—describing each method parameter.

`<paramref>` .NET SHFB S child/inline ● empty   name=*param_name*
Indicates that a word refers to an **ordinary** parameter.

`<permission>` ✔ .NET SHFB S root ● mixed   cref=*member*
Document the security accessibility of a member.
- Very few examples of this tag exist; all set cref to System.Security.PermissionSet.

This wall chart originally appeared in the Simple-Talk.com newsletter and is available electronically here – or type this URL: http://tiny.cc/q9urn.

Tag reference table

Tag	root	child/block	child/inline	namespace	class	constant	field	method	property	event	indexer	operator	constructor	structure	array	interface	enum	delegate	exception	Content	Attributes
`<c>`		●		O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	text	
`<code>`		●		O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	text|element	*many...*
`<example>`	●			O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	mixed	
`<exception>`	●				O	O	O	O	O	O	O	O							O	mixed	cref
`<exclude>`	●			O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	empty	
`<filterpriority>`	●			O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	text	
`<include>`	●			O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	empty	file,path
`<inheritdoc>`	●	●			O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	empty	cref,select
`<list>`		●		O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	element	type
`<note>`			●	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	mixed	type
`<overloads>`	●							O				O								mixed	
`<para>`		●		O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	mixed	
`<param>`	●							A			A	A	A					A		mixed	name
`<paramref>`			●	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	empty	name
`<permission>`	●			O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	mixed	cref
`<pre>`		●		O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	text|element	*many...*
`<preliminary>`	●			O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	mixed	
`<remarks>`	●			O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	mixed	
`<returns>`	●							R	A		A	A						A		mixed	
`<see>`			●	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	mixed	cref|href
`<seealso>`	●			O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	mixed	cref|href
`<summary>`	●			R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	mixed	
`<threadsafety>`				O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	empty	static,instance
`<tocexclude>`	●			O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	empty	
`<typeparam>`	●				A									A						mixed	name
`<typeparamref>`	●			O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	empty	name
`<value>`								R	R											mixed	

R = required   ●   **A** = as needed (i.e. required by context)   ●   **O** = optional

C# Language Specification 4.0 details the language constructs across the top.
Content column terms come from XML standard type declarations.

`<pre>` SHFB child/inline ● text|element
[lang=*language_ID*]   [title=*text*]   [tabsize=*number_of_spaces*]
[numberLines="true"|"false"]   [outlining="true"|"false"]

Syntax-highlights code just like <code> but in HTML additional content files.
- Child: <code>
- <pre> does not apply to XML doc comments, only additional files.
- SHFB adds these custom attributes to the standard HTML <pre> tag.

SHFB Help ► Conceptual and Additional Content ► Additional Content ► Syntax Highlighting <pre> Tag Content

`<preliminary>` SHFB S root ● mixed
Marks the current member as preliminary.
- Empty content generates a default message.

`<remarks>` .NET SHFB S root ● mixed
Generates an eponymous output section.

`<returns>` .NET SHFB S root ● mixed
Describes the return value of a method.

`<see>` ✔ .NET SHFB S child/inline ● mixed   cref=*member* | href=*URL*
Generates an inline hyperlink to an internal (cref) or external (href) target.
- You may use relative (to ./html directory) href paths; same for <a> and <img>.
- See discussions of other important usage points in Taming Sandcastle article.

`<seealso>` ✔ .NET SHFB S root ● mixed   cref=*member* | href=*URL*
Generates an eponymous output section.
- Generates a block hyperlink to an internal (cref) or external (href) target.
- All such links are collected in the "See Also" section of the page.

`<summary>` .NET SHFB S root ● mixed
Generally short description of a type or member at the top of the generated page.
- The most important element; its content is used for Intellisense, quick info, class view, class designer, object browser, as well as aggregate pages in the generated output (e.g. method list page shows <summary> from each method).
- <summary> is required on every member unless inherited with <inheritdoc>.
- Comments automatically inherited for explicit interface members lacking comments.

`<threadsafety>` SHFB S root ● empty static="true"|"false" instance="true"|"false"
Describes how a class or structure behaves in a multi-threaded scenario.
- Specify class references and instance references separately via attributes.

`<tocexclude>` SHFB S root ● empty
Suppresses documenting the current member in the table of contents.

SHFB Help ► Build Process Plug-Ins ► Table of Contents Exclusion Plug-In

`<typeparam>` ✔ .NET SHFB S root ● mixed   name=*type_param_name*
Describes a type parameter of a generic type or method.
- Should be a sentence—not just a fragment—describing each type parameter.

`<typeparamref>` .NET SHFB S child/inline ● empty   name=*type_param_name*
Indicates that a word refers to a **type** parameter.

`<value>` .NET SHFB S root ● mixed
Describes a value of a property.
- Required even though Visual Studio uses <summary>, not <value>, for Intellisense.

Techniques for Including Special XML Characters

(A) Encoding:

◄ ► <	> ► >	& ► &	" ► "	' ► '

(B) By reading from an external file (<include>, <code>).
(C) By using a CDATA section (<![CDATA[. . .]]>).

Material for this wall chart derived in part from:
- Eric Woodruff's Sandcastle Help File Builder Documentation (SHFB Creator)
- C# Prog. Guide: Recommended Tags for Documentation Comments (MSDN)
- Wm. Eric Brunsen's XML Documentation Comments Guide (Dynicity)
- Anson Horton's XML Documentation in C# (Microsoft C# Compiler Program Manager)