

# ADOBE INDESIGN CS4



## ADOBE INDESIGN TAGGED TEXT GUIDE



**JETSET**  
COMMUNICATIONS, INC

TRAINING  
CONSULTING  
SUPPORT



© 2008 Adobe Systems Incorporated and its licensors. All rights reserved.

Using Adobe® InDesign® CS4 Tagged Text for Windows® and Mac OS

If this guide is distributed with software that includes an end user agreement, this guide, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. Except as permitted by any such license, no part of this guide may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written permission of Adobe Systems Incorporated. Please note that the content in this guide is protected under copyright law even if it is not distributed with software that includes an end user license agreement.

The content of this guide is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this guide.

Please remember that existing artwork or images that you may want to include in your project may be protected under copyright law. The unauthorized incorporation of such material into your new work could be a violation of the rights of the copyright owner. Please be sure to obtain any permission required from the copyright owner.

Any references to company names in sample templates are for demonstration purposes only and are not intended to refer to any actual organization.

Adobe, the Adobe logo, InCopy, and InDesign are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Mac OS is a trademark of Apple Inc., registered in the United States and other countries. Windows is a trademark of Microsoft Corporation registered in the U.S and/or other countries. All other trademarks are the property of their respective owners.

Certain Spelling portions of this product are based on Proximity Linguistic Technology. ©Copyright 1990 Merriam-Webster Inc. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2003 Franklin Electronic Publishers Inc. ©Copyright 2003 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. Legal Supplement ©Copyright 1990/1994 Merriam-Webster Inc./Franklin Electronic Publishers Inc. ©Copyright 1994 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990/1994 Merriam-Webster Inc./Franklin Electronic Publishers Inc. ©Copyright 1997 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA ©Copyright 1990 Merriam-Webster Inc. ©Copyright 1993 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2004 Franklin Electronic Publishers Inc. ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1991 Dr. Lluís de Yzaguirre I Maura ©Copyright 1991 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 Munksgaard International Publishers Ltd. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 Van Dale Lexicografie bv ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1995 Van Dale Lexicografie bv ©Copyright 1996 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 IDE a.s. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1992 Hachette/Franklin Electronic Publishers Inc. ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1991 Text & Satz Datentechnik ©Copyright 1991 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2004 Bertelsmann Lexikon Verlag ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2004 MorphoLogic Inc. ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 William Collins Sons & Co. Ltd. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1993-95 Russicon Company Ltd. ©Copyright 1995 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2004 IDE a.s. ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. The Hyphenation portion of this product is based on Proximity Linguistic Technology. ©Copyright 2003 Franklin Electronic Publishers Inc. ©Copyright 2003 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1984 William Collins Sons & Co. Ltd. ©Copyright 1988 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 Munksgaard International Publishers Ltd. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1997 Van Dale Lexicografie bv ©Copyright 1997 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1984 Editions Fernand Nathan ©Copyright 1989 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1983 S Fischer Verlag ©Copyright 1997 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1989 Zanichelli ©Copyright 1989 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1989 IDE a.s. ©Copyright 1989 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 Espasa-Calpe ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1989 C.A. Stromberg AB. ©Copyright 1989 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA.

The Spelling portion of this product is based on Proximity Linguistic Technology. Color-database derived from Sample Books ©Dainippon Ink and Chemicals, Inc., licensed to Adobe Systems Incorporated.

Portions © The Focoltone Colour Systems, and used under license.

This product includes software developed by the Apache Software Foundation ([www.apache.org](http://www.apache.org)).

Portions © 1984-1998 Faircom Corporation. All rights reserved.

Portions copyrighted by Trumatch, Inc. and used under license.

PANTONE® Colors displayed in the software application or in the user documentation may not match PANTONE-identified standards. Consult current PANTONE Color Publications for accurate color. PANTONE® and other Pantone, Inc. trademarks are the property of Pantone, Inc. © Pantone, Inc., 2006

This product contains either BISAFE and/or TIPEM software by RSA Data Security, Inc. Copyright © 1994 Hewlett-Packard Company. Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Hewlett-Packard Company makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Copyright © 1996, 1997 Silicon Graphics Computer Systems, Inc. Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Silicon Graphics makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Video compression and decompression in Adobe® Flash® Player is powered by On2 TrueMotion video technology. © 1992-2005 On2 Technologies, Inc. All rights reserved. <http://www.on2.com>

This product may include software developed by the OpenSymphony Group. (<http://www.opensymphony.com>)

Portions may be licensed from Nellymoser, Inc. (<http://www.nellymoser.com>)

Sorenson Spark™ video compression and decompression technology licensed from Sorenson Media, Inc.

MPEG Layer-3 audio compression technology licensed by Fraunhofer IIS and THOMSON multimedia. (<http://www.iis.fhg.de/amm/>)

Notice to U.S. Government End Users: The Software and Documentation are "Commercial Items," as that term is defined at 48 C.F.R. §2.101, consisting of "Commercial Computer Software" and "Commercial Computer Software Documentation," as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R. §227.7202, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §§227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (a) only as Commercial Items and (b) with only those rights as are granted to all other end users pursuant to the terms and conditions herein. Unpublished-rights reserved under the copyright laws of the United States. Adobe agrees to comply with all applicable equal opportunity laws including, if appropriate, the provisions of Executive Order 11246, as amended, Section 402 of the Vietnam Era Veterans Readjustment Assistance Act of 1974 (38 USC 4212), and Section 503 of the Rehabilitation Act of 1973, as amended, and the regulations at 41 CFR Parts 60-1 through 60-60, 60-250, and 60-741. The affirmative action clause and regulations contained in the preceding sentence shall be incorporated by reference.

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110, USA.

# Contents

## **Adobe InDesign Tagged Text**

Tagged text .....	1
InDesign tags .....	3

# Adobe InDesign Tagged Text

You can use tags in Adobe® InDesign® or Adobe InCopy® to import and export formatting attributes with text-only files. These tags, or codes, indicate style and list definitions as well as the character- and paragraph-level attributes you want to apply to text.

## Tagged text

When you import a tagged text file, InDesign or InCopy reads the tag codes and applies the attributes automatically. When you export formatted text, you can generate the tag codes so that you can later edit the tags and import the information with formatting intact.

<p><b>Anton Guitars</b> Open the case for the <i>first time</i> and take in the sights and smells of your hand-made instrument. See the way the light reflects off the handrubbed satin finish. Breath in the perfume of the finest exotic tone woods. Then, finally, take it in your hands and begin to play.</p>	<pre>&lt;ASCII-MAC&gt; &lt;DefineParaStyle:Head 1=&lt;cTypeface:Plain&gt; &lt;cSize:24.0&gt;&lt;cLeading:18.0&gt;&lt;cFont:Gill Sans&gt;&gt; &lt;ColorTable;=&lt;Black:COLOR:CMYK:Process:0.0000, 0.0000,0.0000,1.0000&gt;&gt; &lt;ParaStyle:Head 1&gt;Anton Guitars &lt;ParaStyle:&gt;&lt;cTypeface:&gt;&lt;cFont:Minion&gt;Open the case for the &lt;cTypeface:&gt;&lt;cFont:&gt; &lt;cTypeface:Italic&gt;&lt;cFont:Minion&gt;first time &lt;cTypeface:&gt;&lt;cFont:&gt;&lt;cTypeface:&gt;&lt;cFont:Minion&gt; and take in the sights and smells of your hand-made instrument. See the way the light reflects off the handrubbed satin finish. Breath in the perfume of the finest exotic tone woods. Then, finally, take it in your hands and begin to play.</pre>
--	--


Formatted InDesign document (left) and exported tagged text file (right), with tags shown in red

Only InDesign and InCopy can read InDesign tags. InDesign cannot read tagged text files formatted in Adobe PageMaker® and QuarkXPress®.

For a more up-to-date version of this Tagged Text PDF, visit [www.adobe.com/go/learn\\_id\\_taggedtext](http://www.adobe.com/go/learn_id_taggedtext).

## Import and export tags

All tags begin with the less-than character (<) and end with the greater-than character (>). Type the tag immediately before the text you want to format. For paragraph-level attributes, type the tag at the start of the paragraph. For character-level attributes, type the tag at the start of the text you want to change. Most type attributes you specify remain in effect until you use codes to cancel them or until you apply other attributes. For example, to apply underlining to one word in a paragraph, you type <cUnderline:1> before the word and <cUnderline:> after the word.

 To learn about using tag codes, export text with tags from a formatted document. Then view the exported text in a word-processing application or text editor to see how the attributes are transformed into tag codes. You can then edit and add tags before you import the tagged text file.

## Export InDesign text as tagged text

- 1 Using the Type tool, click an insertion point in the story you want to export, or select the range of text you want to export.
- 2 Choose File > Export.

- 3 For Save As Type (Microsoft® Windows®) or Format (Apple® Mac OS®), select Adobe InDesign Tagged Text.
- 4 Specify a name and location for the file, and click Save. (If you're saving the file using Mac OS, add a .txt extension to the end of the filename if you want to open the file in Windows.)
- 5 In the Adobe InDesign Tags Export Options dialog box, specify the following options:
  - Select Verbose to show the tags in long form, such as <cStrokeColor:Green> or Abbreviated to show the tags in short form, such as <csc:Green>.
  - Select the encoding format, such as ASCII (the most common format for representing English characters as numbers), ANSI (a collection of many international characters and special punctuation marks that can be inserted using the Alt/Option key), Unicode (a standard that supports most language characters, including non-European languages), Shift-JIS (MS-Kanji for Japanese characters), GB 18030 (the official character set of the People's Republic of China), Big 5 (encoding for Traditional Chinese characters), and KSC5601 (encoding for Korean characters).
- 6 Click OK.

The story text is exported as a text file. You can use a word-processing application or a text editor to open and edit the tagged text file.

### Import tagged text

- 1 Open the document into which you will place the tagged text, and choose File > Place.
- 2 Locate and select the text-only tagged file.
- 3 If desired, select Show Import Options.
- 4 Click Open.
- 5 If you selected Show Import Options in the Place dialog box, select any of the following:

**Use Typographer's Quotes** Ensures that imported text includes left and right quotation marks (“ ”) and apostrophes (') instead of straight quotation marks (" ") and apostrophes (').

**Remove Text Formatting** Removes formatting such as typeface, type color, and type style from the imported text.

**Resolve Text Style Conflicts Using** Lets you specify which character or paragraph style is applied when there is a conflict between the style in the tagged text file and the style in your document. Select Publication Definition to use the definition that already exists for that style name in the document. Select Tagged File Definition to use the style as defined in the tagged text. This option creates another instance of the style with “copy” appended to its name in the Character Style or Paragraph Style panel.

**Show List Of Problem Tags Before Place** Displays a list of unrecognized tags. If a list appears, you can choose to cancel or continue the import. If you continue, the file may not look as expected.

- 6 Click OK.

The text appears at the insertion point or replaces selected text. If there is no text selection or insertion point, a loaded text icon appears. You can click inside an existing frame to place the text inside it, or you can click or drag to create a new text frame.

## Specifying numeric values in tags

Note the following tips for specifying numeric values or text within a tag:

- In the Tag Name column for the following tables, each tag includes the value type. *String* includes a list of characters, such as the name of a color. (When the information is a text string, the case of the characters doesn't matter, unless you're typing a specific color or style name.) *Integer* includes only whole numbers for values, such as the number of drop-cap characters. *Real* can include any number, such as "23.578." *Boolean* includes on/off toggle values: "0" for off and "1" for on. *Enum* includes only specific string values, such as "Small Caps." *Definition* includes valid names of defined styles and lists.
- Numeric measurements are expressed as points.
- If you create a tagged text file in a word-processing application or text editor, any characters, tabs, and spaces you type outside the angle brackets (< >) appear in the document when the file is placed. To begin a new paragraph with no paragraph style assigned, type <ParaStyle:> on a new line.
- To return an attribute to its default state, type the attribute name followed by a colon (:). For example, to return to the default leading value, type <cLeading:>.

## InDesign tags

For a more up-to-date version of these tags, visit [www.adobe.com/go/learn\\_id\\_taggedtext](http://www.adobe.com/go/learn_id_taggedtext).

### Start file and definition tags

When you export text as tagged text, the text file includes a start tag that describes the encoding format (such as <ASCII-WIN>) and a color table tag (<ColorTable>) that defines the colors, tints, and gradients in your document. If you use paragraph and character styles, variables, table and cell styles, or defined lists in exported text, definition tags for those items appear.

**Note:** If you create a tagged text file from scratch, you must include a tag that describes the encoding format (such as <ASCII-MAC>) at the beginning of the text file. You must type the following text at the top of a text file you want InCopy to treat as tagged: <ASCII-WIN> for files created in Windows or <ASCII-MAC> for files created in Mac OS. If you use an encoding format other than ASCII, insert the name of that format in place of ASCII (for example, <UNICODE-MAC>).

Attribute	Format	Examples and notes
Start file tag	<Encodingformat-Platform>	<ASCII-MAC>  Specify the encoding format (ASCII, ANSI, UNICODE, SJIS, CGB18030, BIG5, or KSC5601) followed by the platform (MAC or WIN).
Color table	<ColorTable:=<Swatch name 1:Swatch Type (COLOR, TINT, GRAD, or MIXED):Color Type (CMYK, RGB, LAB, or MIXED):Color Mode (Process, Spot, or Mixed):Values of color, tint, or gradient> <Swatch name 2>...>  A COLOR swatch requires 4 values for CMYK and 3 values for RGB and LAB.  GRADIENT syntax: <Gradient name: GRAD:Fill type:StopPosition1;StopColor1;Midpoint1   StopPosition2;StopColor2;Midpoint2>  TINT syntax: <Base Color:TINT:Tint value>  MIXED Ink Syntax: <MixedInk Name:MIXED:MIXED:Mixed:Component1Name, Component2Name:Component1value,Component2value >	<ColorTable:= <Black:COLOR:CMYK:Process:0,0,0,1> <Green:COLOR:CMYK:Process:1,0,1,0> <Blue:COLOR:CMYK:Process:1,1,0,0> <G/B:GRAD:Linear:0.0;Green;0.5   1.0;Blue;0.5> <Blue:TINT:Blue;20>> <MixedInk1:MIXED:MIXED:Mixed:Green,Blue:1.00,0.5>  The color table is a list of all colors, tints, and gradients defined in the Swatches panel and used in the text.  You can use the abbreviation <Table>.
Paragraph style definition	<DefineParaStyle:Style group name::Paragraph style name= <attr1><attr2>...>  A paragraph style definition includes the character- and paragraph-level attributes listed later in this document, as well as <Nextstyle:style name> and <BasedOn:style name>	<DefineParaStyle:Style Group 1:Heading 1=<Nextstyle:Body Text> <Font:AdobeGaramond> <Size:18> <pBodyAlignment:Center>>  The paragraph style definition includes a list of all styles defined in the Paragraph Styles panel and used in the text. See <a href="#">“Tags for formatting characters and paragraphs”</a> on page 5.
Paragraph style applied	<ParaStyle:Paragraph style name>	<ParaStyle:Heading 1>  The tag <ParaStyle:> starts a new paragraph with no paragraph style. You can use the abbreviation <pStyle>.
Character style definition	<DefineCharStyle:Style group name::Char style name= <char attr1><char attr2>...>  A character style definition includes the character-attributes listed later in this document, as well as <Nextstyle:style name>.	<DefineCharStyle:Style Group 1:Emphasis=<Nextstyle:Emphasis> <Typeface:Italic>>  The character style definition includes a list of all styles defined in the Character Styles panel and used in the text. See <a href="#">“Tags for formatting characters and paragraphs”</a> on page 5.
Character style applied	<CharStyle:Character style name>	<CharStyle:Emphasis>  The tag <CharStyle:> ends the character style definition.
Table style definition	<DefineTableStyle:Style group name:table style name=<table attr1><table attr2>...<cell style>...>  A table style definition includes the table attributes listed later in this document followed by the cell styles used in the table style. The <BasedOn:style name> attributes can also be included.	<DefineTableStyle:Style Group 1:Income Table=<tBeforeSpace:6> <tLastColCellStyle::Blue Fill> <tLastColUseBodyCellStyle:0>>  For information on hyperlink tags, see <a href="#">“Table tags”</a> on page 15.
Table style applied	<TableStyle:Table style name>  The <TableStyle> tag appears before the <TableStart> tag.	<TableStyle:Income Table>

Attribute	Format	Examples and notes
Cell style definition	<p>&lt;DefineCellStyle:Style group name:cell style name=&lt;cell attr 1&gt;&lt;cell attr 2&gt;...&gt;</p> <p>A cell style definition includes the cell attributes listed later in this document, as well as &lt;BasedOn:style name&gt;.</p>	<p>&lt;DefineCellStyle:Style Group 1:Blue Fill:&lt;tCellFillColor:C\=75 M\=0 Y\=0 K\=0&gt;&gt;</p>
Cell style applied	<CellStyle:cell style name>	<p>&lt;CellStyle:Blue Fill&gt;</p> <p>The &lt;CellStyle&gt; tag appears after the &lt;CellStart&gt; tag, and the &lt;End:CellStyle&gt; tag appears after the &lt;CellEnd&gt; tag.</p>
Define list	<p>&lt;DefineListStyle:list name=&lt;list attr 1&gt;&lt;list attr 2&gt;&gt;</p> <p>The two attributes you can include are &lt;ContinuesAcrossStories:0 or 1&gt; and &lt;ContinuesAcrossDocuments:0 or 1&gt; (0=off; 1=on)</p>	<DefineListStyle:Level 1=<ContinuesAcrossStories:0>>
Variable definition	<p>&lt;DefineTextVariable:variable name=&lt;TextVarType:type&gt;&lt;var attr 1&gt;&lt;var attr 2&gt;...&gt;</p> <p>A variable style definition includes the type of variable and variable attributes listed later in this document.</p>	<p>&lt;DefineTextVariable:Chapter Titles=&lt;TextVarType:ChapterNum&gt; &lt;tvTextBefore:Chapter&gt;&gt;</p> <p>TextVarType attributes include CreateDate, ModDate, OutputDate, PageCount, FileName, CustomText, ChapterNum, ParaStyle, and CharStyle</p>
Variable inserted	<TextVarName:var name>	<TextVarName:Chapter Number>
Conditional text definition	<p>&lt;DefineCondition:condition name=&lt;ConditionColor:color&gt; &lt;ConditionIndicatorMethod:Enum&gt; &lt;ConditionIndicatorAppearance:Enum&gt; &lt;ConditionVisibility:Boolean&gt;&gt;</p>	<p>&lt;DefineCondition:Vista=&lt;ConditionColor:0.6,0.4,0&gt; &lt;ConditionIndicatorMethod:Highlight&gt; &lt;ConditionIndicatorAppearance:Solid&gt; &lt;ConditionVisibility:1&gt;&gt;</p> <p>Valid values for &lt;ConditionIndicatorMethod&gt; include Highlight and Underline. Valid values for &lt;ConditionIndicatorAppearance&gt; include Wavy, Solid, and Dashed. For &lt;ConditionVisibility&gt;, 1 is displayed and 0 is hidden.</p>
Conditional text applied	<cConditionalText:condition name>	<cConditionalText:Vista>
Cross-reference format definition	<p>&lt;XRefFormatDefn:=&lt;FormatName:variable name&gt; &lt;CharStyleRef:style name&gt; &lt;BuildingBlocksLength:Integer&gt; &lt;BuildingBlock:=&lt;BlockType:Enum&gt;&gt;&gt;</p> <p>A cross-reference format definition includes the format name, the name of the character style that's applied to the cross-reference source, the number of building blocks used, and building block tags.</p>	<p>&lt;XRefFormatDefn:=&lt;FormatName:Full Paragraph \&amp; Page Number&gt;&lt;CharStyleRef:&gt; &lt;BuildingBlocksLength:4&gt; &lt;BuildingBlock:=&lt;BlockType:CustomString&gt;&lt;CustomText:"&gt;&lt;CharStyleRef:&gt;&lt;IncludeDelim:0&gt;&gt; &lt;BuildingBlock:=&lt;BlockType:FullParagraph&gt;&lt;CustomText:&gt;&lt;CharStyleRef:&gt;&lt;IncludeDelim:0&gt;&gt; &lt;BuildingBlock:=&lt;BlockType:CustomString&gt;&lt;CustomText:" on page &gt;&lt;CharStyleRef:&gt;&lt;IncludeDelim:0&gt;&gt; &lt;BuildingBlock:=&lt;BlockType:PageNumber&gt;&lt;CustomText:&gt;&lt;CharStyleRef:&gt;&lt;IncludeDelim:0&gt;&gt;&gt;</p> <p>For information on hyperlink tags, see <a href="#">"Hyperlink, cross-reference, and index tags"</a> on page 19.</p>
Cross-reference format	XRefFormat:format name	<XRefFormat:Full Paragraph \& Page Number>

## Tags for formatting characters and paragraphs

The following tables describe character- and paragraph-level tags.

## Character-level tags

Attribute	Tag name	Tag abbr.	Examples and notes
Alternate glyph	cAlternateGlyph: <i>Integer</i>	caltg	<cAlternateGlyph:3>  Alternate glyphs for OpenType or Asian fonts such as Tekton Pro MM appear in the Glyphs panel. In this example, 3 represents the third alternate glyph in the pop-up panel.
Auto pair kerning	cAutoPairKern: <i>String</i>	capk	<cAutoPairKern:Optical>  This tag can include Optical or Metrics (default).
Baseline shift	cBaselineShift: <i>Real</i>	cbs	<cBaselineShift:3>  This value can be between -5000 and 5000 points.
Case	cCase: <i>Enum</i>	ccase	<cCase:Small Caps>  Valid values include Small Caps, All Caps, Caps To Small Caps, or Normal (default). Caps To Small Caps is an OpenType font attribute.
Character alignment	cHang: <i>Enum</i>	ch	<cHang:Top>  Valid values include EmTop, EmCenter (default), EmBottom, Baseline, ICFTop, and ICFBottom.
Character skew	cSkew: <i>Real</i>	csk	<cSkew:-25>  The value represents the percentage of character skew from 85 to -85 degrees.
Color: color tint	cColorTint: <i>Real</i>	cct	<cColorTint:80>  Specify a tint percentage.
Color: character fill	cColor: <i>String or Color Definition</i>	cc	<cColor:Green> or <cColor:COLOR:RGB:Process:0.5,1,0>  This tag can include tints, gradients, and unnamed colors. The default color is Black.
Color: character stroke	cStrokeColor: <i>String or Color Definition</i>	csc	<cStrokeColor:Green> or <cColor:COLOR:RGB:Process:0.5,1,0>  This tag can include tints, gradients, and unnamed colors. The default color is None.
Color: fill gradient angle	cGradientAngle: <i>Real</i>	cga	<cGradientAngle:-45>  Angle of linear gradient fill from -180 to 180 degrees.
Color: fill gradient center	cGradientCenter: <i>Real,Real</i>	cgc	<cGradientCenter:-145,-145>  This tag indicates the x and y values of the center point of a radial gradient or the starting point of a linear gradient.
Color: fill gradient length	cGradientLength: <i>Real</i>	cgl	<cGradientLength:2>  This tag indicates the length of a linear gradient ramp or radius of a radial gradient.
Color: stroke gradient angle	cStrokeGradientAngle: <i>Real</i>	csga	<cStrokeGradientAngle:-45>  Angle of linear gradient fill from -180 to 180 degrees.
Color: stroke gradient center	cStrokeGradientCenter: <i>Real,Real</i>	csgs	<cStrokeGradientCenter:-145,-145>  This tag indicates the x and y values of the center point of a radial gradient or the starting point of a linear gradient.

Attribute	Tag name	Tag abbr.	Examples and notes
Color: stroke gradient length	cStrokeGradientLength:Real	csgl	<cStrokeGradientLength:2>  This tag indicates the length of a linear gradient ramp or radius of a radial gradient.
Color: stroke tint	cStrokeTint:Real	cst	<cStrokeTint:80>  Specify a tint percentage.
Color: stroke weight (thickness)	cStrokeWeight:Real	csw	<cStrokeWeight:0.5>  Specify a value between 0 and 800.
Color: stroke alignment	cStrokeAlign:Enum	cssa	<cStrokeAlign:1>  0=Outside, 1=Centered (default)
Miter Limit	cMiterLimit:Real	csml	<cMiterLimit:2.00>  Specify a value between 0 and 500.
Outline Join	cOutlineJoin:Enum	csoj	<cOutlineJoin:1>  0=Miter Join, 1=Round Join, 2=Bevel Join
Conditional text	cConditionalText:String	ccondtxt	<cConditionalText:Vista>  For information on conditional text definitions, see <a href="#">“Start file and definition tags”</a> on page 3.
Font: family	cFont:String	cf	<cFont:Adobe Garamond>  Specify only the family name. To complete the font information, also use <cSize> and <cTypeface>.
Font: point size	cSize:Real	cs	<cSize:18>  Specify a point size between 0.1 and 1296.
Font: typeface	cTypeface:String	ct	<cTypeface:Italic>  The values for this tag depend on which font is used. For example, Times New Roman uses Regular while Helvetica uses Plain.
ILG shift	cILGShift:Real	cilgs	<cILGShift:200>  Specify a value between -8640 and 8640.
Kerning	cKerning:Real	ck	<cKerning:50>
Language	cLanguage:String	clang	<cLanguage:Portuguese: Brazilian>  To use a valid string, type the name of the language as it appears in the Character panel.
Leading	cLeading:Real	cl	<cLeading:18>  This tag lets you specify a leading value between 0 and 5000. To use default auto leading, leave value empty (<cLeading:>).
Leading model	cLeadingModel:Center Down	cleadmod	CenterDown signifies the leading will be measured from the center of the current line to the center of the next line.
Ligatures	cLigatures:Boolean	clig	<cLigatures:1>  1=on (default), 0=off

Attribute	Tag name	Tag abbr.	Examples and notes
No break	cNoBreak: <i>Boolean</i>	cnb	<cNoBreak:1> 1=on, 0=off (default)
Next break character	cNextXChars: <i>Enum</i>	cnxc	<cNextXChars:Column> Valid values include Column, Page, BoxBreak, OddPage, and EvenPage.
Old style figures	cOldStyleFigures: <i>Boolean</i>	cosf	<cOldStyleFigures:1> 1=on, 0=off (default)
OpenType: figure style	cFigureStyle: <i>Enum</i>	cfs	<cFigureStyle:Proportional Oldstyle> Valid values include Tabular Lining (default), Proportional Oldstyle, Proportional Lining, Figure Style, and Tabular Oldstyle.
OpenType: contextual alternate	cOTFContAlt: <i>Boolean</i>	cotfcalt	<cOTFContAlt:0> 1=on (default), 0=off
OpenType: discretionary ligatures	cOTFDiscLig: <i>Boolean</i>	cotfdl	<cOTFDiscLig:1> 1=on, 0=off (default)
OpenType: feature list	cOTFFeatureList: <i>String</i>	cotfl	<cOTFFeatureList:zero,1>
OpenType: fractions	cOTFFractions: <i>Boolean</i>	cotff	<cOTFFractions:1> 1=on, 0=off (default)
OpenType: locale	cOTFLocale: <i>Boolean</i>	cotflocl	<cOTFLocale:1> 0=off (default), 1=on
OpenType: mark position	cOTFMarkPos: <i>Boolean</i>	cotfmark	<cOTFMarkPos:1> 0=off (default), 1=on
OpenType: position	cPosition: <i>Enum</i>	cp	<cPosition:Numerator> Valid values include None (default), Superior, Inferior, Numerator, and Denominator.
OpenType: position form	cOTFPositionForm: <i>Enum</i>	cotfposform	<cOTFPositionForm:Initial> Valid values include Off, Automatic, Initial, Medial, Final, and Isolated.
OpenType:proportional metrics	cOTFUseProportional Metric: <i>Boolean</i>	cotfupm	<cOTFUseProportionalMetric:1> 1=on, 0=off (default)
OpenType:roman italics	cOTFRomanItalics: <i>Boolean</i>	cotfri	<cOTFRomanItalics:1> 1=on, 0=off (default)
OpenType: ordinals	cOTFOrdinal: <i>Boolean</i>	cotfo	<cOTFOrdinal:1> 1=on, 0=off (default)
OpenType: slash zero	cOTFSlashZero: <i>Boolean</i>	cotfsz	<cOTFSlashZero:1> 1=on, 0=off (default)
OpenType: stylistic sets	cOTFStylisticSets: <i>Integer</i>	cotfss	<cOTFStylisticSets:8>

Attribute	Tag name	Tag abbr.	Examples and notes
OpenType: titling alternates	cOTFTitlAlt: <i>Boolean</i>	cotftlt	<cOTFTitlAlt:1> 1=on, 0=off (default)
OpenType:swash alternates	cOTFSwaAlt: <i>Boolean</i>	cotfsalt	cOTFSwaAlt:1> 1=on, 0=off (default)
Overprint character fill	cOverprint: <i>Boolean</i>	co	<cOverprint:1> 1=on, 0=off (default)
Overprint character stroke	cStrokeOverprint: <i>Boolean</i>	cso	<cStrokeOverprint:1> 1=on, 0=off (default)
Prev/Next page number	cPageNumType: <i>Enum</i>	cpnt	<cPageNumType:Next> Valid values include Next, Previous, and Current. Current is the default value.
Rotate Character	cCharRotate: <i>Real</i>	cchr	<cCharRotate:45> Specify a value between -360 and 360 to rotate the character on its center axis. Positive values rotate the character counterclockwise.
Scaling:affects line height	cScaleAffectsLineHeight: <i>Boolean</i>	cslineh	<cScaleAffectsLineHeight:1> 1=on, 0=off (default)
Scaling: horizontal	cHorizontalScale: <i>Real</i>	chs	<cHorizontalScale:2> The value 1.0 equals 100%. You can use values between 1% (.01) and 1000% (10.0).
Scaling: vertical	cVerticalScale: <i>Real</i>	cvs	<cVerticalScale:2> The value 1.0 equals 100%. You can use values between 1% (.01) and 1000% (10.0).
Small cap percentage	cSmallCapPercentage: <i>Real</i>	cscp	<cSmallCapPercentage:70> Specify a percentage value.
Strikethrough	cStrikethru: <i>Boolean</i>	cstrike	<cStrikethru:1> 1=on, 0=off (default)
Strikethrough gap color	cStrikethroughGapColor: <i>Real</i>	cugc	<cStrikethroughGapColor:C\=100 M\=0 Y\=0 K\=0>
Strikethrough gap overprint	cStrikethroughGapOverprint: <i>Boolean</i>	cugo	<cStrikethroughGapOverprint:1> 1=on, 0=off (default)
Strikethrough gap tint	cStrikethroughGapTint: <i>Real</i>	cugt	<cStrikethroughGapTint:30>
Strikethrough offset	cStrikethroughOffset: <i>Real</i>	cuoff	<cStrikethroughOffset:3>
Strikethrough weight offset	cStrikethroughWeightOffset: <i>Real</i>	cuwoff	<cStrikethroughWeightOffset:3>
Strikethrough type	cStrikethroughType: <i>Definition</i>	cutype	<cStrikethroughType:ThickThin>
Superscript or subscript character position	cPosition: <i>Enum</i>	cp	<cPosition:Superscript> Valid values include Superscript, Subscript, or Normal (default).

Attribute	Tag name	Tag abbr.	Examples and notes
Tracking	cTracking: <i>Real</i>	ctk	<cTracking:50> Specify a value between -1000.0 and 10000 (1000ths of an em).
Underline	cUnderline: <i>Boolean</i>	cu	<cUnderline:1> 1=on, 0=off (default)
Underline gap color	cUnderlineGapColor: <i>Real</i>	cugc	<cUnderlineGapColor:C\=100 M\=0 Y\=0 K\=0>
Underline gap overprint	cUnderlineGapOverprint: <i>Boolean</i>	cugo	<cUnderlineGapOverprint:1> 1=on, 0=off (default)
Underline gap tint	cUnderlineGapTint: <i>Real</i>	cugt	<cUnderlineGapTint:30>
Underline offset	cUnderlineOffset: <i>Real</i>	cuoff	<cUnderlineOffset:3>
Underline weight offset	cUnderlineWeightOffset: <i>Real</i>	cuwoff	<cUnderlineWeightOffset:3>
Underline type	cUnderlineType: <i>Definition</i>	cutype	<cUnderlineType:ThickThin>

## Paragraph-level tags

Attribute	Tag name	Tag abbr.	Examples and notes
Absorb ideographic space	pAbsorbIdeoSpace: <i>Boolean</i>	pideosp	<pAbsorbIdeoSpace:1> 1=on, 0=off
Roman word wrap off	pWordWrapOff: <i>Boolean</i>	pwwroff	<pWordWrapOff:1> 1=on, 0=off
Align to baseline grid	pAlignToGrid: <i>Boolean</i>	patg	<pAlignToGrid:1> 1=on, 0=off (default)
Alignment of text in paragraph	pTextAlignment: <i>Enum</i>	pta	<pTextAlignment:Center> Valid values include Left, Right, Center, JustifyLeft, JustifyRight, JustifyCenter, JustifyFull, AwayFromSpine, and ToSpine. Left is the default value.
Balance ragged lines	pBalanceLines: <i>Boolean</i>	pbl	<pBalanceLines:1> 0=off (default), 1=on (Vee shape), 2=Equal, 3=Pyramid The user interface lets you turn balanced lines on or off. The 2 and 3 values are available only for scripting purposes.
Drop cap details	pDropCapDetail: <i>String</i>	pdcdetail	<pDropCapDetail:LeftGlyphEdge\,Descenders> Include LeftGlyphEdge and/or Descenders to turn on Align Left Edge and Scale for Descenders, respectively.
Drop cap: number of characters	pDropCapCharacters: <i>Integer</i>	pdcc	<pDropCapCharacters:1>
Drop cap: number of lines	pDropCapLines: <i>Integer</i>	pdcl	<pDropCapLines:3>
First line indent	pFirstLineIndent: <i>Real</i>	pfli	<pFirstLineIndent:6>
Glyph scaling: desired	pDesiredGlyphScale: <i>Real</i>	pdgs	<pDesiredGlyphScale:1.0> Specify a value between 0.5 (50%) and 2.0 (200%). The default value is 1.0 (100%).

Attribute	Tag name	Tag abbr.	Examples and notes
Glyph scaling: maximum	pMaxGlyphScale: <i>Real</i>	pmaxgs	<pMaxGlyphScale:1.1>  Specify a value between 0.5 (50%) and 2.0 (200%). The default value is 1.0 (100%). The Maximum value must be greater than the Desired value.
Glyph scaling: minimum	pMinGlyphScale: <i>Real</i>	pmings	<pMinGlyphScale:0.9>  Specify a value between 50% (0.5) and 200% (2.0). The default value is 1.0 (100%). The Minimum value must be less than the Desired value.
Hyphenate all capital words	pHyphenateCapitals: <i>Boolean</i>	phc	<pHyphenateCapitals:1>  1=on, 0=off (default)
Hyphenate Last Word	pHyphenateCrossFrame: <i>Boolean</i>	phcf	<pHyphenateCrossFrame:1>  0=off (default), 1=on
Hyphenation: hyphen limit	pHyphenationLadderLimit: <i>Integer</i>	phll	<pHyphenationLadderLimit:5>  Value indicates maximum number of consecutive hyphens from 1 to 25; zero means unlimited.
Hyphenation: last word	pHyphenateLastWord: <i>Boolean</i>	phlw	<pHyphenateLastWord:0>  1=on (default), 0=off
Hyphenation: minimum number of characters after hyphen	pMinCharAfterHyphen: <i>Integer</i>	pmcah	<pMinCharAfterHyphen:3>  Specify a value from 1 to 15. The value 3 is the default.
Hyphenation: minimum number of characters before hyphen	pMinCharBeforeHyphen: <i>Integer</i>	pmcbh	<pMinCharBeforeHyphen:3>  Specify a value from 1 to 15. The value 3 is the default.
Hyphenation: number of characters in shortest word allowed to be hyphenated	pShortestWordHyphenated: <i>Integer</i>	pswh	<pShortestWordHyphenated:3>  This range of this value is from 3 to 25. The default value is 7.
Hyphenation: on or off	pHyphenation: <i>Boolean</i>	ph	<pHyphenation:1>  1=on (default), 0=off
Hyphenation: zone	pHyphenationZone: <i>Real</i>	phz	<pHyphenationZone:48>  Value (in points) is valid only if Single-line Composer is turned on. Specify a value between 0 and 8640. The default value is 36 points.
Ignore Optical Margin Alignment	pIgnoreEdgeAlign: <i>Boolean</i>	piea	<pIgnoreEdgeAlign:1>  0=off (default), 1=on
Keep first n lines together (widow control)	pKeepFirstNLines: <i>Integer</i>	pkfnl	<pKeepFirstNLines:4>  Specify a value from 1 to 50 (1 is equivalent to off).
Keep last lines together (orphan control)	pKeepLastNLines: <i>Integer</i>	pknl	<pKeepLastNLines:2>  Specify a value from 1 to 50 (1 is equivalent to off).
Keep lines together	pKeepLines: <i>Boolean</i>	pkl	<pKeepLines:1>  1=on, 0=off (default)
Keep whole paragraph together	pKeepParaTogether: <i>Boolean</i>	pkpt	<pKeepParaTogether:1>  1=on, 0=off (default)

Attribute	Tag name	Tag abbr.	Examples and notes
Keep with next lines	<code>pKeepWithNext:Integer</code>	pkwn	<code>&lt;pKeepWithNext:4&gt;</code> Specify a value from 0 to 5 (0 means off).
Last line right indent	<code>pLastRightIndent:Real</code>	plri	<code>&lt;pLastRightIndent:6&gt;</code>
Leading: autoleading percentage	<code>pAutoLeadPercent:Real</code>	palp	<code>&lt;pAutoLeadPercent:1.2&gt;</code> The value 1.0 equals 100%. Specify a value between 0 and 5 (500%). The default value is 1.75 (175%) for text in plain text frames and 1 (100%) for text in frame grids.
Left indent	<code>pLeftIndent:Real</code>	pli	<code>&lt;pLeftIndent:6&gt;</code>
Letterspacing: desired	<code>pDesiredLetterspace:Real</code>	pdl	<code>&lt;pDesiredLetterspace:1.0&gt;</code> Specify a value between -1.0 (-100%) and 50 (500%). The default value is 1.0 (100%).
Letterspacing: maximum	<code>pMaxletterspace:Real</code>	pmaxl	<code>&lt;pMaxletterspace:1.2&gt;</code> The value 1.0 equals 100%. Specify a value between the desired letterspacing value and 5 (500%). The default value is 0.
Letterspacing: minimum	<code>pMinLetterspace:Real</code>	pminl	<code>&lt;pMinLetterspace:0.9&gt;</code> The value 1.0 equals 100%. Specify a value between -1 (-100%) and the desired letterspacing value. The default value is 0%.
Nested (run-in) styles	<code>pRunInStyles:Enum</code>	prunin	<code>&lt;pRunInStyles:Character Style 1\,Words\,1\1;Character Style 2\,Forced Line Break\,1\,0\;&gt;</code>  Loop example: <code>&lt;pRunInStyles:Character Style 1\,Words\,1\1;Character Style 2\,Forced Line Break\,1\,0\;\[Repeat]\,Rl loopback\,2,1\;&gt;</code>
Nested line styles	<code>pRunInLineStyle:Enum</code>	pruninLine	<code>&lt;pRunInLineStyle:Character Style 1\,1\;Character Style 2\,1\;&gt;</code>  This tag includes the name of the character style followed by the number of lines it applies to.
GREP styles	<code>pRunInGrepStyles:Enum</code>	pruninGrep	<code>&lt;pRunInGrepStyles:Character Style 1\,\d-\;&gt;</code>  This tag includes the name of the character styles followed by the GREP expression (\d- in this example)
Page or column break before paragraph	<code>pBreakBefore:Enum</code>	pbb	<code>&lt;pBreakBefore:Column&gt;</code>  This tag can include Page, OddPage, EvenPage, Column, or None.
Paragraph rule above color	<code>pRuleAboveColor:String</code>	prac	<code>&lt;pRuleAboveColor:Black&gt;</code>  This value indicates the color for the rule (line) above the paragraph. This string value must be a color name (not an unnamed color definition).
Paragraph rule above gap color	<code>pRuleAboveGapColor:String</code>	pragc	<code>&lt;pRuleAboveGapColor:Blue&gt;</code>
Paragraph rule above keep in frame	<code>pRuleAboveKeepInFrame:Boolean</code>	prakif	<code>&lt;pRuleAboveKeepInFrame:1&gt;</code> 1=on, 0=off (default)

Attribute	Tag name	Tag abbr.	Examples and notes
Paragraph rule above tint	pRuleAboveTint: <i>Real</i>	prat	<pRuleAboveTint:80> This value indicates the tint for the rule (line) above the paragraph. The value must be between 0 and 100.
Paragraph rule above gap tint	pRuleAboveGapTint: <i>Real</i>	pragt	<pRuleAboveGapTint:80>
Paragraph rule above gap overprint	pRuleAboveGapOverprint: <i>Boolean</i>	prago	<pRuleAboveGapOverprint:1> 1=on, 0=off (default)
Paragraph rule above stroke type	pRuleAboveStrokeType: <i>Enum</i>	prastype	<pRuleAboveStrokeType:ThickThin>
Paragraph rule above is turned on	pRuleAboveOn: <i>Boolean</i>	praon	<pRuleAboveOn:1> 1=on, 0=off (default)
Paragraph rule above left indent	pRuleAboveLeftIndent: <i>Real</i>	prali	<pRuleAboveLeftIndent:3> This value indicates the left indent from the frame edge or text edge, depending on the paragraph rule mode.
Paragraph rule above mode	pRuleAboveMode: <i>Enum</i>	pram	<pRuleAboveMode:Text> This value indicates whether the paragraph rule starts at the width of the column (frame) or width of the first line of text in the paragraph. Valid values include Text and Column. Column is the default.
Paragraph rule above offset	pRuleAboveOffset: <i>Real</i>	prao	<pRuleAboveOffset:3> This value indicates the offset from the baseline to the paragraph rule above.
Paragraph rule above right indent	pRuleAboveRightIndent: <i>Real</i>	prari	<pRuleAboveRightIndent:3> This value indicates the right indent from the frame edge or text edge, depending on paragraph rule mode.
Paragraph rule above stroke	pRuleAboveStroke: <i>Real</i>	prast	<pRuleAboveStroke:2> This value indicates the line weight for the paragraph rule above the paragraph. Specify a value between 0 and 1000.
Paragraph rule below color	pRuleBelowColor: <i>String</i>	prbc	<pRuleBelowColor:Black> This value indicates the color for the rule (line) below the paragraph. This string value must be a color name (not an unnamed color definition).
Paragraph rule below gap color	pRuleBelowGapColor: <i>String</i>	prbgc	<pRuleBelowGapColor:Blue>
Paragraph rule below tint	pRuleBelowTint: <i>Real</i>	prbt	<pRuleBelowTint:80> This value indicates the tint for the rule (line) below the paragraph. The value must be between 0 and 100.
Paragraph rule below gap tint	pRuleBelowGapTint: <i>Real</i>	prbgt	<pRuleBelowGapTint:80>
Paragraph rule below gap overprint	pRuleBelowGapOverprint: <i>Boolean</i>	prbgo	<pRuleBelowGapOverprint:1> 1=on, 0=off (default)
Paragraph rule below stroke type	pRuleBelowStrokeType: <i>Enum</i>	prbstype	<pRuleBelowStrokeType:ThickThin>

Attribute	Tag name	Tag abbr.	Examples and notes
Paragraph rule below is turned on	pRuleBelowOn: <i>Boolean</i>	prbon	<pRuleBelowOn:1> 1=on, 0=off (default)
Paragraph rule below left indent	pRuleBelowLeftIndent: <i>Real</i>	prbli	<pRuleBelowLeftIndent:3> This value indicates the left indent from the frame edge or text edge, depending on paragraph rule mode.
Paragraph rule below mode	pRuleBelowMode: <i>Enum</i>	prbm	<pRuleBelowMode:Text> This value indicates whether the paragraph rule starts at the width of the column (frame) or width of the first line of text in the paragraph. Valid values include Text and Column. Column is the default.
Paragraph rule below offset	pRuleBelowOffset: <i>Real</i>	prbo	<pRuleBelowOffset:3> This value indicates the offset from the bottom of the embox of the largest character in the line to the paragraph rule below.
Paragraph rule below right indent	pRuleBelowRightIndent: <i>Real</i>	prbri	<pRuleBelowRightIndent:3> This value indicates the right indent from the frame edge or text edge, depending on paragraph rule mode.
Paragraph rule below stroke	pRuleBelowStroke: <i>Real</i>	prbs	<pRuleBelowStroke:2> This value indicates the line weight for the paragraph rule below the paragraph.
Paragraph rule below tint	pRuleAboveTint: <i>Real</i>	prbt	<pRuleAboveTint:80> This value indicates the tint for the rule (line) below the paragraph. The value must be between 0 and 100.
Right indent	pRightIndent: <i>Real</i>	pri	<pRightIndent:6>
Space after paragraph	pSpaceAfter: <i>Real</i>	psa	<pSpaceAfter:4>
Space before paragraph	pSpaceBefore: <i>Real</i>	psb	<pSpaceBefore:4>
Tab ruler settings	pTabRuler: <i>x value, tab type, align on, default, leader;</i>	ptr	<pTabRuler:48,Center,,0;;96,Char,x,0,*;144,Left,,0;;> A semicolon (;) indicates the start of a new tab setting. The <i>x value</i> determines the distance from the left edge of the text frame or inset in horizontal frames (top edge in vertical frames). <i>Tab type</i> includes Left, Center, Right, and Char (also called a decimal tab). <i>Align on</i> determines the character on which a decimal tab (Char) will align. <i>Default</i> is a boolean value (0 or 1) that indicates whether the tab is a default tab setting. <i>Leader</i> is a character or string of characters (such as periods or hyphens) you want to use as a tab leader.
Text composition engine	pTextComposer: <i>String</i>	ptc	<pTextComposer:Adobe Single-Line Composer> Specify any composer available, including the Paragraph Composer (default) and the Single-line Composer.
Word spacing: align single word	pSingleWordAlignment: <i>Enum</i>	pswa	<pSingleWordAlignment:Right>

Attribute	Tag name	Tag abbr.	Examples and notes
Word spacing: desired	pDesiredWordSpace:Real	pdws	<pDesiredWordSpace:1.2> Specify a value between 0 and 10 (1000%). The default value is 1.0 (100%).
Word spacing: maximum	pMaxWordSpace:Real	pmaws	<pMaxWordSpace:1.2> The value 1.0 equals 100%. The default value is 1.3 (133%). Specify a value between 0 and 10 (1000%) not less than Desired value.
Word spacing: minimum	pMinWordSpace:Real	pmiws	<pMinWordSpace:0.9> The value 1.0 equals 100%. The default value is 0.8 (80%). Specify a value between 0 and 10 (1000%) not greater than Desired value.

## Table tags

For information on table and cell style definition tags, see “[Start file and definition tags](#)” on page 3.

Attribute	Tag name	Tag abbr.	Examples and notes
Table start	TableStart:Real,Real, Real, Real	tStart	<tStart:5,4, 1, 0> <tStart: rows, columns, header rows, footer rows>
Table end	TableEnd	tEnd	<tEnd>
Row start	RowStart	rStart	<rStart>
Row end	RowEnd	rEnd	<rEnd>
Cell start	CellStart:Real,Real	cStart	<cStart:1,1>
Cell end	CellEnd	cEnd	<cEnd>
Column start	ColStart	coStart	<coStart>
Start headers	tStartHeadersOn:Real	tshon	<tStartHeadersOn:2> Possible values are 0, 1, 2 0 = Repeat header on every text column 1 = Repeat header once per frame 2 = Repeat header once per page
Start footers	tStartFootersOn:Real	tsfon	<StartFootersOn:1> Possible values are 0, 1, 2 0 = Repeat footer on every text column 1 = Repeat footer once per frame 2 = Repeat footer once per page
Skip first header	tSkipFirstHeader:Boolean	tsfh	<tSkipFirstHeader:1> 1=on, 0=off (default)
Skip first footer	tSkipFirstFooter:Boolean	tsff	<tSkipFirstFooter:1> 1=on, 0=off (default)
Cell: bottom inset	tCellAttrBottomInset:Real	tcabi	<tCellAttrBottomInset:9>
Cell: bottom stroke color	tCellBottomStrokeColor:Definition	tcbsc	<tCellBottomStrokeColor:Blue>
Cell: bottom stroke tint	tCellAttrBottomStrokeTint:Real	tcabst	<tCellAttrBottomStrokeTint:70>

Attribute	Tag name	Tag abbr.	Examples and notes
Cell: bottom stroke gap tint	tCellBottomStrokeGapTint: <i>Real</i>	tcbsgt	<tCellBottomStrokeGapTint:95>
Cell: bottom stroke type	tcBottomStrokeType: <i>Enum</i>	tcbst	<tcBottomStrokeType:ThinThick>
Cell: bottom stroke gap color	tCellbottomStrokeGapColor: <i>Real</i>	tcbsgc	<tCellBottomStrokeGapColor:C\=100 M\=0 Y\=0 K\=0>
Cell: bottom stroke gap overprint	tCellBottomStrokeGapOverprint: <i>Boolean</i>	tcbsgo	<tCellBottomStrokeGapOverprint:0> 1=on, 0=off (default)
Cell: default cell type	tCellDefaultCellType: <i>Enum</i>	tc dct	<tCellDefaultCellType:1>
Cell: fill color	tCellFillColor: <i>Definition</i>	tcfc	<tCellFillColor:Blue>
Cell: fill tint	tCellAttrFillTint: <i>Real</i>	tcaft	<tCellAttrFillTint:20>
Cell: first line offset	tTextCellFirstLineOffset: <i>Enum</i>	tcflo	<tTextCellFirstLineOffset:2> Valid values include 1 (Ascent), 2 (Cap Height), 3 (Leading), 4 (x Height), and 5 (Fixed). 1 is the default.
Cell: left inset	tCellAttrLeftInset: <i>Real</i>	tcali	<tCellAttrLeftInset:9>
Cell: left stroke color	tCellLeftStrokeColor: <i>Definition</i>	tclsc	<tCellLeftStrokeColor:Blue>
Cell: left stroke tint	tCellAttrLeftStrokeTint: <i>Real</i>	tcalst	<tCellAttrLeftStrokeTint:70>
Cell: left stroke type	tcLeftStrokeType: <i>Enum</i>	tclst	<tcLeftStrokeType:ThinThick>
Cell: left stroke gap tint	tCellleftStrokeGapTint: <i>Real</i>	tclsgt	<tCellLeftStrokeGapTint:95>
Cell: left stroke gap color	tCellLeftStrokeGapColor: <i>Real</i>	tclsgc	<tCellLeftStrokeGapColor:C\=100 M\=0 Y\=0 K\=0>
Cell: left stroke gap overprint	tCellLeftStrokeGapOverprint: <i>Boolean</i>	tclsgo	<tCellLeftStrokeGapOverprint:0> 1=on, 0=off (default)
Cell: outer bottom stroke color	tCellOuterBottomStrokeColor: <i>Definition</i>	tcobsc	<tCellOuterBottomStrokeColor:Blue>
Cell: outer bottom stroke type	tOuterBottomStrokeType: <i>Enum</i>	tobst	<tOuterBottomStrokeType:ThinThick>
Cell: outer left stroke color	tCellOuterLeftStrokeColor: <i>Definition</i>	tc colsc	<tCellOuterLeftStrokeColor:Blue>
Cell: outer left stroke type	tOuterLeftStrokeType: <i>Enum</i>	tolst	<tOuterLeftStrokeType:ThinThick>
Cell: outer right stroke color	tCellOuterRightStrokeColor: <i>Definition</i>	tcorsc	<tCellOuterRightStrokeColor:Blue>
Cell: outer right stroke type	tOuterRightStrokeType: <i>Enum</i>	torst	<tOuterRightStrokeType:ThinThick>
Cell: outer top stroke color	tCellOuterTopStrokeColor: <i>Definition</i>	tcotsc	<tCellOuterTopStrokeColor:Blue>
Cell: outer top stroke type	tOuterTopStrokeType: <i>Enum</i>	totst	<tOuterTopStrokeType:ThinThick>
Cell: right inset	tCellAttrRightInset: <i>Real</i>	tcari	<tCellAttrRightInset:9>
Cell: right stroke color	tCellRightStrokeColor: <i>Definition</i>	tc rsc	<tCellRightStrokeColor:Blue>
Cell: right stroke tint	tCellAttrRightStrokeTint: <i>Real</i>	tc arst	<tCellAttrRightStrokeTint:70>

Attribute	Tag name	Tag abbr.	Examples and notes
Cell: right stroke type	tRightStrokeType:Enum	tcrst	<tRightStrokeType:ThinThick>
Cell: right stroke gap tint	tCellRightStroke GapTint:Real	tcrsgt	<tCellRightStrokeGapTint:95>
Cell: right stroke gap color	tCellRightStroke GapColor:Real	tcrsgc	<tCellRightStrokeGapColor:C\=100 M\=0 Y\=0 K\=0>
Cell: right stroke gap overprint	tCellRightStrokeGap Overprint:Boolean	tcrsgo	<tCellRightStrokeGapOverprint:0> 1=on, 0=off (default)
Cell: rotation	tCellAttrRotation:Enum	tcar	<tCellAttrRotation:180> Valid values include 0 (default), 90, 180, and 270.
Cell: stroke weight bottom	tCellAttrBottom StrokeWeight:Real	tcabsw	<tCellAttrBottomStrokeWeight:3>
Cell: stroke weight left	tCellAttrLeft StrokeWeight:Real	tcalsw	<tCellAttrLeftStrokeWeight:3>
Cell: stroke weight right	tCellAttrRight StrokeWeight:Real	tcarsw	<tCellAttrRightStrokeWeight:3>
Cell: stroke weight top	tCellAttrTop StrokeWeight:Real	tcatsw	<tCellAttrTopStrokeWeight:3>
Cell: top inset	tCellAttrTopInset:Real	tcati	<tCellAttrTopInset:9>
Cell: top stroke color	tCellTopStrokeColor:Definition	tctsc	<tCellTopStrokeColor:Blue>
Cell: top stroke tint	tCellAttrTopStrokeTint:Real	tcatst	<tCellAttrTopStrokeTint:70>
Cell: top stroke type	tTopStrokeType:Enum	tctst	<tTopStrokeType:ThinThick>
Cell: top stroke gap tint	tCellTopStrokeGapTint:Real	tctsgt	<tCellTopStrokeGapTint:95>
Cell: top stroke gap color	tCellTopStrokeGapColor:Real	tctsgc	<tCellTopStrokeGapColor:C\=100 M\=0 Y\=0 K\=0>
Cell: top stroke gap overprint	tCellTopStrokeGap Overprint:Boolean	tctsgo	<tCellTopStrokeGapOverprint:0> 1=on, 0=off (default)
Cell: type	tCellType:Enum	tct	<tCellType:1>
Cell: vertical composition	tTextCellVerticalComposition:Enum	tcvc	<tTextCellVerticalComposition:1>
Cell: vertical justification	tTextCellVertical Justification:Enum	tcvj	<tTextCellVerticalJustification:2> Valid values include 0 (Top), 1 (Bottom), 2 (Center), and 3 (Justify)
Cell: vertical justification paragraph spacing limit	tTextCellAttrMaxVJ InterParaSpace:Real	ttcamvjips	<tTextCellAttrMaxVJInterParaSpace:40>
Column width	tColAttrWidth:Real	tcaw	<tColAttrWidth:120>
Column: fill pattern (first color)	tColFillPatFirstColor:Enum	tcfpcl	<tColFillPatFirstColor:ThinThick>
Column: fill pattern (second color)	tColFillPatSecondColor:Enum	tcfpscl	<tColFillPatSecondColor:ThinThick>
Column: stroke pattern (first color)	tColStrokePatFirstColor:Enum	tcspfl	<tColStrokePatFirstColor:ThinThick>

Attribute	Tag name	Tag abbr.	Examples and notes
Column: stroke pattern (first type)	tColStrokePattern FirstType:Enum	tcsplt	<tColStrokePatternFirstType:ThinThick>
Column: stroke pattern (second color)	tColStrokePat SecondColor:Enum	tcrpscl	<tColStrokePatSecondColor:ThinThick>
Column: stroke pattern (second type)	tColStrokePattern SecondType:Enum	tcsplt	<tColStrokePatternSecondType:ThinThick>
Column: stroke pattern first weight	tColStrokePattern FirstWeight:Real	tcsplw	<tColStrokePatternFirstWeight:9>
Column: stroke pattern second weight	tColStrokePattern SecondWeight:Real	tcspsw	<tColStrokePatternSecondWeight:9>
Column: stroke type	tColStrokeType:Enum	tcst	<tColStrokeType:ThickThin>
Row height	tRowAttrHeight:Real	trah	<tRowAttrHeight:20>
Row: AutoGrow	tRowAutoGrow:Enum	trag	<tRowAutoGrow:1>
Row: fill pattern (first color)	tRowFillPatFirstColor:Enum	trfpfcl	<tRowFillPatFirstColor:ThinThick>
Row: fill pattern (second color)	tRowFillPat SecondColor:Enum	trfpscl	<tRowFillPatSecondColor:ThinThick>
Row: row size maximum	tRowAttrMaxRowSize:Real	tramars	<tRowAttrMaxRowSize:40>
Row: row size minimum	tRowAttrMinRowSize:Real	tramirs	<tRowAttrMinRowSize:9>
Row: stroke pattern (first color)	tRowStrokePat FirstColor:Enum	trspfcl	<tRowStrokePatFirstColor:ThinThick>
Row: stroke pattern (first type)	tRowStrokePattern FirstType:Enum	trspft	<tRowStrokePatternFirstType:ThinThick>
Row: stroke pattern (second color)	tRowStrokePat SecondColor:Enum	trpscl	<tRowStrokePatSecondColor:ThinThick>
Row: stroke pattern (second type)	tRowStrokePattern SecondType:Enum	trspst	<tRowStrokePatternSecondType:ThinThick>
Row: stroke pattern first weight	tRowStrokePattern FirstWeight:Real	trspfw	<tRowStrokePatternFirstWeight:9>
Row: stroke pattern second weight	tRowStrokePattern SecondWeight:Real	trpsw	<tRowStrokePatternSecondWeight:9>
Row: stroke type	tRowStrokeType:Enum	trst	<tRowStrokeType:ThickThin>
Table: column end pattern (start value)	tTableColFillPattern EndValue:Real	trcfpev	<tTableColFillPatternEndValue:1>
Table: column fill pattern (first count)	tTableColFillPattern FirstCount:Real	trcfpc	<tTableColFillPatternFirstCount:2>
Table: column fill pattern (second count)	tTableColFillPattern SecondCount:Real	trcfpsc	<tTableColFillPatternSecondCount:2>
Table: column fill pattern (start value)	tTableColFill PatternStartValue:Real	trcfpsv	<tTableColFillPatternStartValue:1>
Table: column stroke color	tColStrokeColor:Definition	tcsc	<tColStrokeColor:Blue>
Table: column stroke pattern (first count)	tTableColStrokePattern FirstCount:Real	tcspfc	<tTableColStrokePatternFirstCount:2>

Attribute	Tag name	Tag abbr.	Examples and notes
Table: column stroke pattern (second count)	tTableColStrokePattern SecondCount: <i>Real</i>	tcspsc	<tTableColStrokePatternSecondCount:2>
Table: column stroke weight	tColStrokeWeight: <i>Real</i>	tcsww	<tColStrokeWeight:3>
Table: fill pattern priority	tTableFillPatternPriority: <i>Real</i>	tfpp	<tTableFillPatternPriority:1>
Table: outer bottom stroke tint	tOuterBottomStrokeTint: <i>Real</i>	tobst	<tOuterBottomStrokeTint:70>
Table: outer bottom stroke weight	tOuterBottomStrokeWeight: <i>Real</i>	tobsw	<tOuterBottomStrokeWeight:3>
Table: outer left stroke tint	tOuterLeftStrokeTint: <i>Real</i>	tolst	<tOuterLeftStrokeTint:70>
Table: outer left stroke weight	tOuterLeftStrokeWeight: <i>Real</i>	tolsw	<tOuterLeftStrokeWeight:3>
Table: outer right stroke tint	tOuterRightStrokeTint: <i>Real</i>	torst	<tOuterRightStrokeTint:70>
Table: outer right stroke weight	tOuterRightStrokeWeight: <i>Real</i>	torsw	<tOuterRightStrokeWeight:3>
Table: outer top stroke tint	tOuterTopStrokeTint: <i>Real</i>	totst	<tOuterTopStrokeTint:70>
Table: outer top stroke weight	tOuterTopStrokeWeight: <i>Real</i>	totsw	<tOuterTopStrokeWeight:3>
Table: row end pattern (start value)	tTableRowFillPatternEndValue: <i>Real</i>	trfpev	<tTableRowFillPatternEndValue:1>
Table: row fill pattern (first count)	tTableRowFillPatternFirstCount: <i>Real</i>	trfpfc	<tTableRowFillPatternFirstCount:2>
Table: row fill pattern (second count)	tTableRowFillPatternSecondCount: <i>Real</i>	trfpfc	<tTableRowFillPatternSecondCount:2>
Table: row fill pattern (start value)	tTableRowFillPatternStartValue: <i>Real</i>	trfpsv	<tTableRowFillPatternStartValue:1>
Table: row keeps	tRowKeeps: <i>Real</i>	trk	<tRowKeeps:3>
Table: row stroke color	tRowStrokeColor: <i>Definition</i>	trsc	<tRowStrokeColor:Blue>
Table: row stroke pattern (first count)	tTableRowStrokePatternFirstCount: <i>Real</i>	trspfc	<tTableRowStrokePatternFirstCount:2>
Table: row stroke pattern (first count)	tTableRowStrokePatternSecondCount: <i>Real</i>	trspfc	<tTableRowStrokePatternSecondCount:2>
Table: row stroke weight	tRowStrokeWeight: <i>Real</i>	trsw	<tRowStrokeWeight:3>
Table: stroke drawing order	tStrokeOrder: <i>Real</i>	tso	<tStrokeOrder:3>  Possible values are 0-3: 0 - Best Join 1 - Row Strokes in Front 2 - Column Strokes in Front 3 - InDesign 2.0 Compatability
Table: spacing after	tAfterSpace: <i>Real</i>	tas	<tAfterSpace:9>
Table: spacing before	tBeforeSpace: <i>Real</i>	tbs	<tBeforeSpace:9>

## Hyperlink, cross-reference, and index tags

For information on cross-reference format tags, see “[Start file and definition tags](#)” on page 3.

Attribute	Tag name	Tag abbr.	Examples and notes
Hyperlink	Hyperlink: <i>Definition</i>	hpl	<Hyperlink:=<HyperlinkName:Page 3>>
Hyperlink destination definition	HyperlinkDestDefn: <i>Definition</i>	hpldestdfn	<HyperlinkDestDefn:=<HyperlinkName:Page 3>>
Hyperlink name	HyperlinkName: <i>Real</i>	hplname	<HyperlinkName:Page 3>
Hyperlink destination	HyperlinkDest: <i>Real</i>	hpldest	<HyperlinkDest:Chapter 1>
Hyperlink destination key	HyperlinkDestKey: <i>Integer</i>	DeskKey	<HyperlinkDestKey:12>
Hyperlink length	HyperlinkLength: <i>Real</i>	hpllen	<HyperlinkLength:6> Specify the number of characters in the hyperlink.
Hyperlink destination name	HyperlinkDestName: <i>Real</i>	hpldestname	<HyperlinkDestName:Page 3>
Hyperlink destination URL	HyperlinkDestUrl: <i>Real</i>	hpldesturl	<HyperlinkDestUrl:http://www.adobe.com>
Hyperlink destination index	HyperlinkDestIndex: <i>Real</i>	hpldestidx	<HyperlinkDestIndex:1>
Hyperlink destination page	HyperlinkDestPage: <i>Real</i>	hpldestpage	<HyperlinkDestPage:3>
Hyperlink destination page zoom type	HyperlinkDestPageZoomType: <i>Real</i>	hpldestpagezoomtype	<HyperlinkDestPageZoomType:Fixed> Valid values include Fixed, Fit View, Fit in Window, Fit Width, Fit Height, Fit Visible, and Inherit Zoom.
Hyperlink destination page zoom factor	HyperlinkDestPageZoomFactor: <i>Real</i>	hpldestpagezoomfactor	<HyperlinkDestPageZoomFactor:1.7> 1=100% magnification
Hyperlink length	HyperlinkLength: <i>Integer</i>	HplLen	<HyperlinkLength:33> Specifies the number of characters, including spaces, in the hyperlink source
Hyperlink start offset	HyperlinkStartOffset: <i>Real</i>	HplOff	<HyperlinkOffset:0>
Hyperlink destination paragraph	IsParagraphDest: <i>Boolean</i>		<IsParagraphDest:1> 1=paragraph destination; 0=other hyperlink destination type, such as text anchor
Hyperlink hidden	Hidden: <i>Boolean</i>	hid	<Hidden:0>
Hyperlink border width	BrdrWidth: <i>Real</i>	brdrw	<BrdrWidth:2>
Hyperlink border visible	BrdrVisible: <i>Boolean</i>	brdrv	<BrdrVisible:1>
Hyperlink border highlight	BrdrHilight: <i>Boolean</i>	brdrh	<BrdrHilight:1>
Hyperlink border style	BrdrStyle: <i>Enum</i>	brdrs	<BrdrStyle:1>
Hyperlink border color	BrdrColor: <i>Real</i>	brdrc	<BrdrColor:Blue>
Hyperlink cross-reference building block	BuildingBlock:=<BlockType: <i>Enum</i>	BldBlk	<BuildingBlock:=<BlockType:PageNumber><CustomText:><CharStyleRef:Red><IncludeDelim:0>> Valid building blocks include CustomString, FileName, ChapterNumber, PageNumber, FullParagraph, ParagraphNumber, ParagraphText, and TextAnchorName.
Hyperlink cross-reference character style	CharStyleRef: <i>Real</i>	CharStyleRef	<CharStyleRef:Red> Specify the name of the character style applied to the cross-reference building block.

Attribute	Tag name	Tag abbr.	Examples and notes
Hyperlink cross-reference include delimiter	IncludeDelim: <i>Boolean</i>	InclDim	<IncludeDelim:1>  1=include delimiter in character style; 0=don't include (default)
Index entry	IndexEntry	Idx	<IndexEntryType:IndexPageEntry> <IndexEntryStyleUid:0><IndexEntryRangeType:kCurrentPage><IndexEntryDisplayString:simple page index>>
Index entry type	IndexEntryType	IdxEnType	<IndexEntryType:IndexPageEntry>
Index entry range type	IndexEntryRangeType	IdxEnRngType	<IndexEntryRangeType:kForNextNParagraphs>
Index entry style	IndexEntryStyle	IdxEnStyle	<IndexEntryStyle:MyStyle>
Index entry next style	IndexEntryNextStyle	IdxEnNxtStyle	<IndexEntryNextStyle:MyStyle>
Index entry number of paragraph or page	IndexEntryNumParOrPage	IdxEnNumPgOrPar	<IndexEntryNumParOrPage:30>
Index entry display string	IndexEntryDisplayString	IdxEnDispStr	<IndexEntryDisplayString:simple page index>
Index entry sorting string	IndexEntrySortString	IdxEnSortStr	<IndexEntrySortString:An index>
Index cross-reference entry type	IndexXRefEntryType	IdxXrEnType	<IndexXRefEntryType:kSeeHerein>
Index cross-reference entry custom string	IndexXRefEntryCustomString	IdxXrEnCusStr	<IndexXRefEntryCustomString:abc>

## Footnote tags

Attribute	Tag name	Tag abbr.	Examples and notes
Footnote start	FootnoteStart	fnStart	<FootnoteStart:>
Footnote end	FootnoteEnd	fnEnd	<FootnoteEnd:>

## Bullets and numbering tags

The following InDesign CS2 tags are no longer available in InDesign CS4: bnSize (bnsz), bnColor (bnc), bnSeparator (bnsep), numFont (nmf), and numTypeFace (nmtf). These tags are imported in InDesign CS4 as text.

Attribute	Tag name	Tag abbr.	Examples and notes
List type	bnListType: <i>Enum</i>	bnlt	<bnListType:Bullet>  Valid values include Bullet and Numbered.
Numbering style for bullets and numbering	bnNumberingStyle: <i>Enum</i>	bnns	<bnNumberingStyle:01\,02\,03\,04...>  Valid values include the items that can be selected from the Format menu, including 1\, 2\, 3\, 4... and a\, b\, c\, d...
Number start at	bnNumberStartAt: <i>Integer</i>	bnnsa	<bnNumberStartAt:7>
Bullet character	bnBulletCharacter: <i>Unicode setting,String</i>	bnbc	<bnBulletCharacter:UnicodeOnly\,9674>  Valid values can be UnicodeOnly or UnicodeWithFont followed by the unicode value.

Attribute	Tag name	Tag abbr.	Examples and notes
Number expression	numNumber: <i>String</i>	nmnum	<numNumber:^H.^#> Valid strings include the characters and number placeholders inserted in the Number field.
List level	numListLevel: <i>Enum</i>	nmll	<numListLevel:2> Valid integers range from 1 to 9.
Bullet alignment	bulAlignment: <i>Enum</i>	bla	<bulAlignment:Center> Valid values include Left, Right, and Center.
Number alignment	numAlignment: <i>Enum</i>	nma	<numAlignment:Center> Valid values include Left, Right, and Center.
Bullet character style used	bulCharStyle: <i>Definition</i>	blcs	<bulCharStyle:Emphasis> Valid values include character style names in the current document.
Number character style used	numCharStyle: <i>Definition</i>	nmcs	<bulCharStyle:Emphasis> Valid values include character style names in the current document.
Restart numbering	numShouldRestart: <i>Boolean</i>	nmsr	<numShouldRestart:1> 1=restart numbering; 0=don't restart
Continue numbering from previous	numCFPrevious: <i>Boolean</i>	nmcfp	<numCFPrevious:1> 1=continues numbering from previous; 0=don't continue numbering
List name	numListName: <i>Definition</i>	nmln	<numListName:Level 1> Valid values include defined list names in the current document.
Restart policy	numRestartPolicy: <i>Enum</i>	nmrp	<numRestartPolicy:LevelRange\2\4> Valid values include LevelRange ("2\4" restarts levels 2 through 4) and Specific Level (\3\3" restarts numbers after level 3).
Text after bullet	bulTextAfter: <i>String</i>	blta	<bulTextAfter:\^t>

## Variable tags

For information on the variable definition tag, see [“Start file and definition tags”](#) on page 3.

Attribute	Tag name	Tag abbr.	Examples and notes
Text Before	tvTextBefore: <i>String</i>	tvtb	<tvTextBefore:Chapter>
Text After	tvTextAfter: <i>String</i>	tvta	<tvTextAfter: out of>
Date Format	tvDateFormat: <i>Enum</i>	tvdf	<tvDateFormat:ddMMMMyyyy> For valid formats, see the table in the text variables section of help.
Custom text string	tvString: <i>String</i>	tvsv	<tvString: Maria Sanchez>

Attribute	Tag name	Tag abbr.	Examples and notes
Numbering style	tvNumStyle:Enum	tvnst	<tvNumStyle:SmallLetter> Valid values include SmallLetter, LargeLetter, SmallRoman, LargeRoman, and Arabic. If not specified, the current numbering style is used.
Scope	tvScope:Enum	tvsc	<tvScope:Document> Valid values include Document and Section.
Folder path	tvIncFolderPath:Boolean	tvifp	<tvFolderPath:1> (1=include path; 0=don't include path)
File extension	tvFileExtn:Boolean	tvfe	<tvFileExtn:1> (1=include extension; 0=don't include extension)
Delete end punctuation	tvDelEndPunc:Boolean	tvdep	<tvDelEndPunc:1> (1=delete end punctuation; 0=don't delete)
Change case	tvChangeCase:Enum	tvcc	<tvChangeCase:LowerCase> Valid values include LowerCase, UpperCase, TitleCase, and SentenceCase. If not specified, case is not changed.
Style	tvStyle:Definition	tvst	<tvStyle:Head1> Valid values include paragraph and character style names in the current document.
Use first or last style on page	tvUse:Enum	tvu	<tvUse:FirstInPage> Valid values include FirstInPage and LastInPage.

## Tags for special characters

If you want to add special characters into a tagged text file, type the following values into the tagged text file:

Special character	Value
Paragraph return (hard return)	<0x000D>
Line break (soft return)	<0x000A>
Auto page numbering	<0xE0018>
Section marker	<0xE0019>
Bullet character	<0x2022>
Copyright symbol	<0x00A9>
Degree symbol	<0x00B0>
Ellipsis	<0x2026>
Paragraph symbol	<0x00B6>
Registered trademark symbol	<0x00AE>
Section symbol	<0x00A7>
Trademark symbol	<0x2122>
Em dash	<0x2014>

Special character	Value
En dash	<0x2013>
Em space	<0x2003>
Ideographic space	<0x3000>
En space	<0x2002>
Flush space	<0x2001>
Hair space	<0x200A>
Sixth space	<0x2006>
Quarter space	<0x2005>
Third space	<0x2004>
Punctuation space	<0x2008>
Figure space	<0x2007>
Nonbreaking space	<0x00A0>
Nonbreaking space (fixed width)	<0x202F>
Thin space	<0x2009>
Discretionary hyphen	<0x00AD>
Nonbreaking hyphen	<0x2011>
Double left quotation mark	<0x201C>
Double right quotation mark	<0x201D>
Single left quotation mark	<0x2018>
Single right quotation mark	<0x2019>
Non-joiner	<0x200C>

## Tags for InDesign ME

Attribute	Tag name	Tag abbr.	Examples and notes
Hindi digits	cHindiDigits:Enum	chindi	<cHindiDigits:2> 0=default, 1=Arabic, 2=hindi, 3=farsi
Direction override	cDirOverride:Enum	cdirov	<cDirOverride:1> 0=default, 1=LTR, 2=RTL
Diacritic vertical position	cDiacVPos:Enum	cdvpos	<cDiacVPos:2> 0=Off, 1=Loose, 2=Medium, 3=Right, 4=OpenType
Kashida	cKashida:Boolean	ckash	<cKashida:1> 0=off (default), 1=on
Paragraph direction	pParaDir:Boolean	ppdir	<pParaDir:1> 0=left to right, 1=right to left
Diacritic X offset	cDiacXOffset:Real	cddx	<cDiacXOffset:30>

<b>Attribute</b>	<b>Tag name</b>	<b>Tag abbr.</b>	<b>Examples and notes</b>
Diacritic Y offset	cDiacYOffset: <i>Real</i>	cddy	<cDiacYOffset:30>
Overlap swash	cOverlapSwach: <i>Boolean</i>	cOvlapSw	<cOverlapSwach:1> 0=off, 1=on Note: "Swach" was inadvertently misspelled.
Stylistic alternative	cStylisticAlt: <i>Boolean</i>	cStyAlt	<cStylisticAlt:1> 0=off, 1=on
Justification alternative	cJustifAlt: <i>Boolean</i>	cJusAlt	<cJustifAlt:1> 0=off, 1=on
Keyboard direction override	cKbdDirOverride: <i>Enum</i>	ckbdir	<cKbdDirOverride:1> 0=default, 1=LTR, 2=RTL
Justification method	pJustifMethod: <i>Enum</i>	pjmeth	<pJustifMethod:2> 0=default, 1=Arabic, 2=Naskh

