BRADY
WHEN PERFORMANCE MATTERS MOST™

Brady Printer Language

Programmer's Manual

BRADY

This page is intentionally left blank.

Copyright and Trademarks

Disclaimer

This manual is proprietary to **Brady** Worldwide, Inc. (hereafter "**Brady**"), and may be revised from time to time without notice. **Brady** disclaims any understanding to provide you with such revisions, if any.

This manual is copyrighted with all rights reserved. No portion of this manual may be copied or reproduced by any means without the prior written consent of **Brady**.

While every precaution has been taken in the preparation of this document, **Brady** assumes no liability to any party for any loss or damage caused by errors or omissions or by statements resulting from negligence, accident, or any other cause. **Brady** further assumes no liability arising out of the application or use of any product or system described, herein; nor any liability for incidental or consequential damages arising from the use of this document. **Brady** disclaims all warranties of merchantability of fitness for a particular purpose.

Brady reserves the right to make changes without further notice to any product or system described herein to improve reliability, function, or design.

Trademarks

All brand or product names referenced in this manual are trademarks (™) or registered trademarks (®) of their respective companies or organizations.

© 2013 **Brady Corporation**. All Rights Reserved.

Version 1.1

Brady Corporation
6555 West Good Hope Road
P.O. Box 2131
Milwaukee, WI 53201-2131
www.bradycorp.com

Sales/Customer Support:(800) 537-8791

Overview

About this Document

The purpose of this document is to define the Brady Printer Language (BPL), which is a printer control language that will be used by Brady printers.

The Brady Printer Language provides programmatic label definition and printer controls for Brady printers. In addition to interpreters for other scripting languages (ZPL, jScript, etc.) which are currently used by many of Brady's OEM customers, the Brady Printer Language is resident on the printer. This allows Brady's customers to use their existing label scripts (in other languages) with the Brady printer, and write new scripts using the BPL language.

Version

This table describes the features included with each version of this manual.

Version	Available Commands
1.0	Initial release.
1.1	Program editing, revisions

Contents

Copyright and Trademarks iii
Overview iv

BPL Programming Language

Introduction 1-1
Overview 1-1
XML Basics 1-2
XML Elements 1-2
Before you Begin 1-4
BPL Elements 1-5
BPL Hierarchy and Organization 1-6
Structure Hierarchy 1-6
Script with Descriptions 1-7
Create Star Script 1-8
Match Media Supply Script 1-8
Job Control Elements 2-1
<?xml version="1.0"?> 2-1
<bpl-document> 2-1
<labels> 2-1
<!-- --> 2-2
Default Elements 2-3
<defaults> 2-3
<document> 2-3
<printer> 2-4
Label Elements 2-5
<label> 2-5
<rectangle> 2-8
<circle> 2-10
<not> 2-12
<star> 2-14
<triangle> 2-16
<seal> 2-18
<cross> 2-20
<line> 2-22
<arrow> 2-23
<text> 2-26
<barcode> 2-34
<graphic> 2-36
BPL Glossary A-1

1 BPL Programming Language

Introduction

Brady Print language (BPL) is a simple-to-use, plain text programming language. With BPL's human-readable language, you can easily and quickly create labels rather than spending a lot of time developing scripts that use complicated and cryptic syntax.

Overview

The BPL Language is an xml-based printer control language used for generating labels without using any labeling software. Extensible Markup Language (XML) emphasizes simplicity, generality, and usability. It uses a human-readable format with a general data structure, strict syntax and hierarchical structure. This document provides the structure and syntax for the BPL programming language and assumes a basic understanding of xml programming.

Your Brady printer will only recognize and process scripts with the .xml extension.

Note: if you are sending a BPL script from a PC, the extension depends on the application being used.

It is strongly recommended that an XML editor is used to edit files. This ensures that the schema file can be used to validate the script before sending it to the printer. While you can create a script without an XML editor, it is likely that script errors will exist and the printer will fail to print as expected. These errors will not be caught if using an editor that does not support schema files.

XML Basics

To start a script, copy and paste the following script into the XML document:

```
<?xml version="1.0" encoding="UTF-8"?>
<bpl-document xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.bradycorp.com/printers/bpl"
xmlns:bpl="http://www.bradycorp.com/printers/bpl"
xsi:schemaLocation="http://www.bradycorp.com/printers/bpl
enter_local_path_to_file/BPL.xsd">
</bpl-document>
```

Replace the text "**enter_local_path_to_file**" with the path where the bpl.xsd file is located on your computer.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<bpl-document xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.bradycorp.com/printers/bpl"
xmlns:bpl="http://www.bradycorp.com/printers/bpl"
xsi:schemaLocation=http://www.bradycorp.com/printers/bpl C:/BPL/BPL.xsd>
</bpl-document>
```

XML Elements

XML documents are designed in the form of a tree, starting with a root node (bpl-document), and branching to leaves (elements). The end of each element must be identified. You can identify the end of an element in two ways.

1. On a separate line, inside brackets (< >), enter a forward slash (/) followed by the element name.

Example: Element name - star

```
<star height="2.0" width="2.0" line-thickness="4">
</star>
```

2. You can also end the element by simply placing a forward slash (/) before the final bracket (>) in the element code line. You do not have to place the ending code on a separate line.

Example: Element name - star

```
<star height="2.0" width="2.0" line-thickness="4" />
```

It is important that only one end mark be used for each element. If multiple end marks exist, the script will not print correctly.

The following list of (free) editors can be used with the BPL schema file. This list is for your convenience only. We do not recommend any one editor over the others, and are not responsible for any problems related to using these editors. If desired, you can also purchase commercially available XML editors.

Name	Windows	Mac OS X	Linux	Current Web Page
Serna Free XML Editor	Y	Y	Y	http://www.syntext.com/products/serna-free/
XML Copy Editor	Y	N	Y	http://xml-copy-editor.sourceforge.net/
XML Notepad	Y	N	N	http://xmlnotepad.codeplex.com/
XMLPad	Y	N	N	http://www.wmhelp.com/xmlpad3.htm

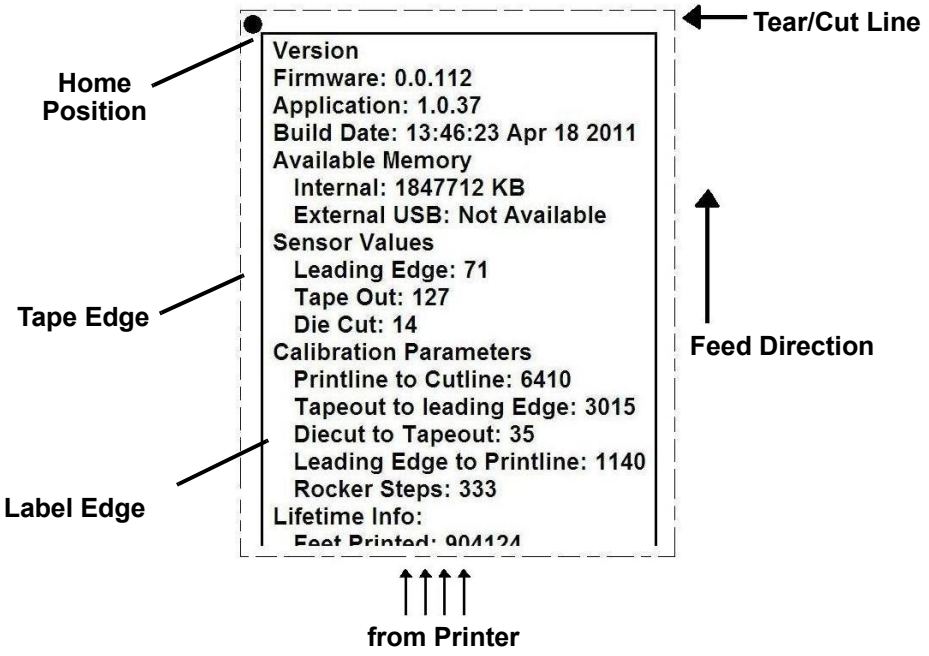
Note: Before creating complex scripts, thoroughly review XML concepts.

You can access the schema file in the Documentation folder on the Product CD in the subfolder called **_BPL Schema** or on the following website: http://www.bradyid.com/bradyid/downloads/downloadsDetailView.do/~0/~0/~0/US_WIDEN_Brady_Printer_Language_Manual.pdf/~0/~0/~0/~0/~0/~0.html.

Before you Begin

Before starting, you should be aware of the following:

- Print resolution indicates dots per inch (dpi), so, the resolution of your printer determines the size of your object.
- The starting (home) position for all objects is always the upper-left corner of the label (see graphic).



- For all "float" types, if *Units* attribute is set to dots, any decimal value will be ignored (i.e., if value is set to 22.6, it will be interpreted as 22).
- For linear dimensional units, all attributes that specify linear distances use the default units that are currently active unless they are explicitly changed in the script file. Options for distance units are: *inches*, *millimeters* or *dots*.

BPL Elements

Chapter 2 contains tables that describe the structure and formats for the elements used in the BPL programming language. The tables follow the same basic structure needed to create a BPL script. An example script follows:

Example

The following example shows a BPL script that produces the label shown.

Note: The BPL namespace defines the types used in this script. In this example, the declared BPL namespace is the default namespace for the entire script (http://www.bradycorp.com/printers/bpl).

Note: If desired, you can use whitespace to make the script more readable.

```
<?xml version = "1.0" ?>
<bpl-document xmlns = "http://www.bradycorp.com/printers/bpl" >
   <defaults>
      <document units = "inches" />
   </defaults>
   <labels>
      <label
            width  = "2.0"
            height = "3.0">
         <barcode
            position-x = "0.1"
            position-y = "0.2"
            height = "1.0"
            type = "code 39"
            human-readable = "true"
            human-readable-location = "bottom" >
         <datasource>
            <static-text value = "Y123456" />
         </datasource>
         </barcode>
      </label>
   </labels>
</bpl-document>
```

BPL Hierarchy and Organization

The BPL Elements tables in Chapter 2 give examples of the structure needed for that *specific* element only. In all cases, the script should follow the structure and hierarchy as shown in *Figure 2*.

Note: The *"Examples" on page 1-7* gives an example BPL script with a description of the elements and syntax used.

Structure Hierarchy

The following example shows the basic structure and hierarchy needed for all BPL scripts:

```
<?xml version="1.0" ?>
<bpl-document xmins = "http://www.bradycorp.com/printers/bpl" >
  <!—This is a comment and is ignored by the program.-->
    <defaults>
      <document units+"inches" />
      <default-label width="100" height="300"...
      <printer tear-or-cut="none".../>
    </defaults>
    <labels>
      <diagnostic-label print="xxx" /> (options are font-list, graphic-list or config-setting)
      <label width="200" height="300"...>
        <rectangle position-x="5.0" height="2.0"... /> (object definition details)
        <text position-x="5.0" position-y+"5.0"... />
          <text-sizing>
            <free-form  height="15.0" /> (options are autosize, free-form or manual)
          </text-sizing>
          <datasource>
            <static-text value="Enter" default="Text" /> (datasource option)
          </datasource>
        </text>
      </label>
    </labels>
</bpl-document>
```

Figure 2. BPL Structure Hierarchy

Examples

Script with Descriptions

The basic structure of a BPL file consists of a start (**<bpl-document>**) and end (**</bpl-document>**) tag. All of the content for the script is contained inside these start- and end- element tags.

Note: When editing the actual text of a BPL script file, indenting is helpful to organize related elements and maintain a human-readable form.

The following shows an example of a basic BPL script:

Elements/Syntax	Description
<?xml version="1.0"?>	*Declaration statement and first line of any script.*
<bpl-document xmins = "http://www.bradycorp.com/printers/bpl">	*Indicates start of a new BPL script.*
<defaults>	*Start of defaults section.*
<default-label *xxx*="x" *yyy*="y" ... />	*Sets default label attributes for the entire job.*
<printer *xxx*="x" />	*Sets default printer attributes for the entire job.*
<document units="x" />	*Sets default document units for the entire job.*
</defaults >	*Ends default section.*
<labels>	*Start of labels container for all other elements.*
<label *xxx*="x" *yyy*="y" ...>	*Start of label container.*
<*object1 xxx*="x" *yyy*="y"... >	*Starts an object element (e.g., rectangle, text, circle, etc.).*
<datasource>	*Start of datasource element for static text.*
<static-text value="xxx" />	*Static text value.*
</datasource>	*End of datasource element.*
</*object1*>	*End of object element.*
<*object2*... *xxx*="x" *yyy*="y"... />	*Object-specific element.*
</label>	*End of label container.*
</labels>	*End of labels container.*
</bpl-document>	*End of BPL script.*

Create Star Script

The following shows an example of a typical label script to create a star.

```
<?xml version="1.0"?>
<bpl-document>
   <defaults>
      <!-- Defines default settings for printer and all labels.-->
      <document units="inches" />
      </defaults>
   <labels>
      <label>
         <star position-x="2" position-y="2" height="2.0"
         width="2.0" line-thickness="4"/>
      </label>
   </labels>
</bpl-document>
```

Match Media Supply Script

The following shows an example of a script that specifies the use of a specific ribbon and tape media. The value specified must exactly match the name printed on the media label.

```
<?xml version="1.0"?>
<bpl-document>
   <defaults>
      <!-- Specifies the specific media type to use for label.-->
      <printer match-media="Y123456" match-ribbon="Y1223344" />
      </defaults>
   <labels>
      <label>
         <text position-x="2" position-y="2" align="left" font-
         name="Arial" >
            <text-sizing>
               <autosize height="50.0" width="50.0" />
            </text-sizing>
            <datasource>
               <static-text value="Print tape/ribbon number." />
            </datasource>
         </text>
      </label>
   </labels>
</bpl-document>
```

2 BPL Elements

Job Control Elements

Job Control elements are used to manage print jobs. All elements require an "end" element.

Element: <?xml version="1.0"?>

Syntax: <?xml version="1.0"?>

Description: *Root element required at start of every script.*

Element: <bpl-document>

Syntax: <bpl-document>

Description: *Indicates the start-of-job element. First element in a BPL script file and indicates the start of a new print job. Only one <bpl-document> element is allowed per script file.*

</bpl-document>

Description: *Indicates the end-of-job element. Last element in the code issued in a BPL script file. Script processing halts when a </bpl-document> command is received.*

Element: <labels>

Syntax: <labels>

Description: *Indicates start of label definitions. This element can contain one or more label or diagnostic-label elements. Only one labels element is allowed within a BPL file.*

</labels>

Description: *Indicates the end of the label definitions.*

Element: **<!-- -->**

Syntax: <!-- *comment-text*

Description: *Indicates start of a comment. Comments continue until an end-comment tag is received. Comments can be on the same line as a BPL element, however, they* **cannot** *be embedded within an element (they must come after the end-element tag). To force a line return within a comment, press Return key.*

-->

Description: *Indicates end of a comment. End tag (-->) can be placed on a separate line after the comment text, or at the end of the last text string.*

Type: string

Default Elements

Default elements set the defaults for the printer and labels. These defaults are used to print all labels unless they are specifically changed in the script. The default element must be placed before the labels element.

Element: **<defaults>**

Syntax:	<defaults>	**Description**: *Child of <bpl-document> element. Indicates the start of the default definitions for the entire job.*
	</defaults>	**Description**: *Indicates the end of the default definitions for the entire job.*

Child Element:	**<document>**	**Description**: *Child of <defaults> element. Specifies default document values. Default is inches.*
Attribute:	**units**	**Description**: *Sets the default units for the printer.*
		Note: Print speed is calculated based on the unit of measure.
Name/Value Pair:	units=*"inches"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	

Child Element:	**<printer>**	**Description**: *Child of <defaults> element. Specifies the default printer settings.*
Attribute:	**match-media**	**Description**: *Sets the media serial number that should be matched to this label job. User is given the option to print anyway, using the installed media.*
Name/Value Pair:	match-media=*"Y123456"*	
Values:	*mediaNumber*	**Options**: *mediaNumber*
Type:	string	
	match-ribbon	**Description**: *Sets the ribbon serial number that should be matched to this label job.*
Name/Value Pair:	match-ribbon=*"Y122334"*	
Values	*ribbonNumber*	**Options**: *ribbonNumber*
Type:	string	
Attribute:	**tear-or-cut-between**	**Description**: *Sets whether the printer will advance each label/row to the tearbar before going to the next label/row. If set, printer feeds each label/row to the tearbar before proceeding to the next label/row.*
Name/Value Pair:	tear-or-cut= *"after-job"*	
Values:	*after-job/between-labels/none*	**Options**: *after-job, between-labels, never.*
Type:	boolean	
Attribute:	**heat**	**Description**: *Sets printhead energy setting, with lowest energy setting at 1. Printhead energy setting is printer dependent (see your printer manual for settings).*
Name/Value Pair:	heat=*"5"*	
Units:	*-10.../...+ 10*	**Options**: *-10 to +10*
Type:	integer	
Attribute:	**speed**	**Description**: *Sets print speed of printer (calculated based on the printer unit of measure). If none is specified, ips is assumed.*
Name/Value Pair:	speed=*"2"*	
Units:	*ips/mps/dps*	**Options**: *inches per second (ips), millimeters per second (mps), dots per second (dps)*
Type:	float	

Label Elements

Label elements define the label content (objects) and attributes.

Element:	**<label>**	**Description**: *Start of definition for label content. Child of <labels> element . Specifies values for the following label (these values override the default-label settings).*
Attribute:	**width**	**Description**: *Sets width from the starting position of the current label.*
Name/Value Pair:	width=*"4"*	**Note:** Height and width are **not** dependent on label orientation (i.e., portrait or landscape). Crossweb (width) = x, Downweb (height) = y. Must be a positive value (see your printer manual for maximum values).
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**height**	**Description**: *Sets height of current label (overrides default). Must be a positive value (see your printer manual for maximum value).*
Name/Value Pair:	Height=*"2"*	**Note:** Height and width are **not** dependent on label orientation (i.e., portrait or landscape). Crossweb (width) = x, Downweb (height) = y. Must be a positive value (see your printer manual for maximum values).
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**offset-x**	**Description**: *Sets label offset (horizontal position) from the edge of the label material.*
Name/Value Pair:	offset-x=*".03"*	
Units:	*inches/millimeters/dots-x*	**Options**: *Any positive or negative number*
Type:	float	
Attribute:	**offset-y**	**Description**: *Sets label offset (vertical position) from the starting position.*
Name/Value Pair:	offset-y=*".06"*	
Units:	*inches/millimeters/dots-y*	**Options**: *Any positive or negative number*
Type:	float	
Attribute:	**orientation**	**Description**: *Sets orientation of image on the label.*
Name/Value Pair:	orientation=*"portrait"*	
Values:	*portrait/landscape*	**Options**: *portrait, landscape*
Type:	string	

Element:	**<label>** *(cont'd)*	**Description**: *Start of definition for label content. Child of <labels> element . Specifies values for the following label (these values override the default-label settings).*
Attribute:	**tear-bar-adjustment**	**Description**: *Sets a change of the transport distance from the printhead to tearbar (in inches/millimeters/dots). Can be positive or negative. See printer manual for details.*
Name/Value Pair:	tear-bar-adjustment=*"1"*	
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**copies**	**Description**: *Sets the number of copies to print for this label.*
Name/Value Pair:	copies=*"2"*	
Values:	*copies*	**Options**: *copies*
Type:	integer	
Attribute:	**font-name**	**Description**: *Sets the default font for the label only. Fonts are printer dependent (see your printer manual for a list of installed fonts).*
Name/Value Pair:	font-name=*"Arial"*	
Values:	*fontName*	**Options**: *fontName* **Note:** If the font name is misspelled, the text will be printed with the printer-specific default font.
Type:	string	
Attribute:	**font-size**	**Description**: *Sets the font size for the label only.*
Name/Value Pair:	font-size=*"12"*	
Units:	*points*	**Options**: *points*
Type:	float	
Attribute:	**label-negative**	**Description**: *Sets inverse printing on label (e.g., white on black becomes black on white).*
Name/Value Pair:	label-negative=*"false"*	
Values:	*true/false*	**Options**: *true/false*
Type:	boolean	

Element:	**<label>** *(cont'd)*	**Description**: *Start of definition for label content. Child of <labels> element . Specifies values for the following label (these values override the default-label settings).*
Attribute:	**label-mirror**	**Description**: *Flips the label image horizontally along the X axis (overrides default setting). For continuous supply, flips the image from the upper left corner of the label (x). Diecuts TBD.*
Name/Value Pair:	label-mirror=*"false"*	
Values:	*true/false*	**Options**: *true, false*
Type:	boolean	
Attribute:	**label-rotate**	**Description**: *Rotates image clockwise by the specified number of degrees.*
Name/Value Pair:	label-rotate=*"90"*	
Units:	*deg*	**Options**: *0 - 359 degrees.*
Type:	integer	

Element:	**<rectangle>**	**Description**: *Child of <label> element. Creates a rectangle.*
Attribute:	**position-x**	**Description**: *Required. Sets position (horizontal displacement), as measured from the home position.*
Name/Value Pair:	position-x=*".03"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**position-y**	**Description**: *Required. Sets position (vertical displacement), as measured from the home position.*
Name/Value Pair:	position-y=*".06"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**height**	**Description**: *Required. Sets shape height (vertical distance), as measured from starting position of the shape – must be positive,*
Name/Value Pair:	height=*"2"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**width**	**Description**: *Required. Sets shape width (horizontal distance), as measured from starting position of the shape – must be positive.*
Name/Value Pair:	width=*"4"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**rotation**	**Description**: *Sets amount of clockwise rotation (number of degrees) to apply to the shape.*
Name/Value Pair:	rotation=*"90"*	
Units:	*deg*	**Options**: *0-359*
Type:	integer	

Element:	**<rectangle>** *(cont'd)*	**Description**: *Child of <label> element. Creates a rectangle.*
Attribute:	**line-thickness**	**Description**: *Sets shape line-width, as measured in printable dots.*
Name/Value Pair:	line-thickness=*"4"*	
Units:	*dots*	**Options**: *1-100 (dots)*
Type:	float	
Attribute:	**fill**	**Description**: *Sets shape background to 100% solid fill.*
Name/Value Pair:	fill=*"none"*	
Values:	*none/solid*	**Options**: *none/solid*
Type:	string	
Attribute:	**rounded-corners**	**Description**: *Sets how much rounding (radius) should be applied to shape corners. Ignored if a shape that does not apply is selected.*
Name/Value Pair:	rounded-corners=*"2"*	
Values:	*1/2/3/4/5/6/7/8*	**Options**: *1-8.*
Type:	integer	

Element:	**<circle>**	**Description**: *Child of <label> element. Creates a circle (or an ellipse if the height and width are different values).* ○
Attribute:	**position-x**	**Description**: *Required. Sets position (horizontal displacement) for the start of the (invisible) box in which the circle is contained, as measured from the home position.*
Name/Value Pair:	position-x=*".03"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**position-y**	**Description**: *Required. Sets position (vertical displacement), for the start of the (invisible) box in which the circle is contained, as measured from the home position.*
Name/Value Pair:	position-y=*".06"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**height**	**Description**: *Required. Sets shape height (vertical distance), as measured from starting position of the shape.* **Note:** If height and width are different values, the result will be an ellipse.
Name/Value Pair:	height=*"2"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**width**	**Description**: *Required. Sets shape width (horizontal distance), as measured from starting position of the shape.* **Note:** If height and width are different values, the result will be an ellipse.
Name/Value Pair:	width=*"2"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**rotation**	**Description**: *Sets amount of clockwise rotation (number of degrees) to apply to an ellipse.*
Name/Value Pair:	rotation=*"0"*	
Units:	*deg*	**Options**: *0-359*
Type:	integer	

Element:	**<circle>** *(cont'd)*	**Description**: *Child of <label> element. Creates a circle (or an ellipse if the height and width are different values).* ◯
Attribute:	**line-thickness**	**Description**: *Sets shape line-width, as measured in printable dots.*
Name/Value Pair:	line-thickness=*"4"*	
Units:	*dots*	**Options**: *1-100 dots*
Type:	float	
Attribute:	**fill**	**Description**: *Sets shape background filling or shading.*
Name/Value Pair:	fill=*"none"*	
Values:	*none/solid*	**Options**: *none/solid*
Type:	string	

Element:	**<not>**	**Description**: *Child of <label> element. Creates a not sign.* ∅
Attribute:	**position-x**	**Description**: *Required. Sets position (horizontal displacement) as measured from the home position.*
Name/Value Pair:	position-x=*".03"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**position-y**	**Description**: *Required. Sets position, (vertical displacement) as measured from the home position.*
Name/Value Pair:	position-y=*".06"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**height**	**Description**: *Required. Sets shape height (vertical distance), as measured from starting position of the shape.*
Name/Value Pair:	height=*"2"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**width**	**Description**: *Required. Sets shape width (horizontal distance), as measured from starting position of the shape.*
Name/Value Pair:	width=*"4"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**rotation**	**Description**: *Sets number of degrees of clockwise rotation to apply to the shape.*
Name/Value Pair:	rotation=*"0"*	
Units:	*deg*	**Options**: *0-359*
Type:	integer	

Element:	**<not>** *(cont'd)*	**Description**: *Child of <label> element. Creates a not sign.* ∅
Attribute:	**line-thickness**	**Description**: *Sets shape line-width, as measured in printable dots.*
Name/Value Pair:	line-thickness=*"4"*	
Units:	*dots*	**Options**: *1-100 dots.*
Type:	float	
Attribute:	**fill**	**Description**: *Sets shape background filling/shading.*
Name/Value Pair:	fill=*"none"*	
Values:	*none/solid*	**Options**: *none/solid*
Type:	string	

Element:	**<star>**	**Description**: *Child of <label> element. Creates a star shape.*
Attribute:	**position-x**	**Description**: *Required. Sets position (horizontal displacement) for the start of the (invisible) box in which the star is contained, as measured from the home position.*
Name/Value Pair:	position-x=*".03"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**position-y**	**Description**: *Required. Sets position (vertical displacement) for the start of the (invisible) box in which the star is contained, as measured from the home position.*
Name/Value Pair:	position-y=*".06"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**height**	**Description**: *Required. Sets shape height (vertical distance), as measured from starting position of the shape.*
Name/Value Pair:	height=*"3"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**width**	**Description**: *Required. Sets shape width (horizontal distance), as measured from starting position of the shape.*
Name/Value Pair:	width=*"3"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**rotation**	**Description**: *Sets amount of clockwise rotation (number of degrees) to apply to the shape.*
Name/Value Pair:	rotation=*"0"*	
Units:	*deg*	**Options**: *0-359*
Type:	integer	

Element:	**<star>** *(cont'd)*	**Description**: *Child of <label> element. Creates a star shape.*
Attribute:	**line-thickness**	**Description**: *Sets shape line-width, as measured in printable dots.*
Name/Value Pair:	line-thickness=*"4"*	
Units:	*dots*	**Options**: *1-100 dots.*
Type:	float	
Attribute:	**fill**	**Description**: *Sets shape background filling/shading.*
Name/Value Pair:	fill=*"none"*	
Values:	*none/solid*	**Options**: *none/solid*
Type:	string	

Element:	**<triangle>**	**Description**: *Child of <label> element. Creates a triangle.*
Attribute:	**position-x**	**Description**: *Required. Sets position (horizontal displacement) for the start of the (invisible) box in which the triangle is contained, as measured from the home position.*
Name/Value Pair:	position-x=*".03"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**position-y**	**Description**: *Required. Sets position (vertical displacement) for the start of the (invisible) box in which the triangle is contained, as measured from the home position.*
Name/Value Pair:	position-y=*".06"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**height**	**Description**: *Required. Sets shape height (vertical distance), as measured from starting position of the shape.*
Name/Value Pair:	height=*"4"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**width**	**Description**: *Required. Sets shape width (horizontal distance), as measured from starting position of the shape.*
Name/Value Pair:	width=*"2"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**rotation**	**Description**: *Sets amount of clockwise rotation (number of degrees) to apply to the shape.*
Name/Value Pair:	rotation=*"0"*	
Units:	*deg*	**Options**: *0-359*
Type:	integer	

Element:	**<triangle>** *(cont'd)*	**Description**: *Child of <label> element. Creates a triangle.*
Attribute:	**line-thickness**	**Description**: *Sets shape line-width, as measured in printable dots.*
Name/Value Pair:	line-thickness=*"4"*	
Units:	*dots*	**Options**: *1-100 dots.*
Type:	float	
Attribute:	**fill**	**Description**: *Sets shape background filling/shading.*
Name/Value Pair:	fill=*"none"*	
Values:	*none/solid*	**Options**: *none/solid*
Type:	string	

Element:	**<seal>**	**Description**: *Child of <label> element. Creates a seal.*
Attribute:	**position-x**	**Description**: *Required. Sets position (horizontal displacement) for the start of the (invisible) box in which the seal is contained, as measured from the home position.*
Name/Value Pair:	position-x=*".03"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**position-y**	**Description**: *Required. Sets position (vertical displacement) for the start of the (invisible) box in which the seal is contained, as measured from the home position.*
Name/Value Pair:	position-y=*".06"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**height**	**Description**: *Required. Sets shape height (vertical distance), as measured from starting position of the shape.*
Name/Value Pair:	height=*"2"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**width**	**Description**: *Required. Sets shape width (horizontal distance), as measured from starting position of the shape.*
Name/Value Pair:	width=*"4"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**rotation**	**Description**: *Sets amount of clockwise rotation (number of degrees) to apply to the shape.*
Name/Value Pair:	rotation=*"0"*	
Units:	*deg*	**Options**: *0-359*
Type:	integer	

Element:	**<seal>** *(cont'd)*	**Description**: *Child of <label> element. Creates a seal.*
Attribute:	**line-thickness**	**Description**: *Sets shape line-width, as measured in printable dots.*
Name/Value Pair:	line-thickness=*"4"*	
Units:	*dots*	**Options**: *1-100 dots.*
Type:	float	
Attribute:	**fill**	**Description**: *Sets shape background filling/shading.*
Name/Value Pair:	fill=*"none"*	
Values:	*none/solid*	**Options**: *none/solid*
Type:	string	

Element:	**<cross>**	**Description**: *Child of <label> element. Creates a cross.*
Attribute:	**position-x**	**Description**: *Required. Sets position (horizontal displacement) for the start of the (invisible) box in which the cross is contained, as measured from the home position.*
Name/Value Pair:	**position-x=***".03"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**position-y**	**Description**: *Required. Sets position (vertical displacement) for the start of the (invisible) box in which the cross is contained, as measured from the home position.*
Name/Value Pair:	position-y=*".06"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**height**	**Description**: *Required. Sets shape height (vertical distance), as measured from starting position of the shape.*
Name/Value Pair:	height=*"4"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**width**	**Description**: *Required. Sets shape width (horizontal distance), as measured from starting position of the shape.*
Name/Value Pair:	width=*"2"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**rotation**	**Description**: *Sets amount of clockwise rotation (number of degrees) to apply to the shape.*
Name/Value Pair:	rotation=*"0"*	
Units:	*deg*	**Options**: *0-359*
Type:	integer	

Element:	**<cross>** *(cont'd)*	**Description**: *Child of <label> element. Creates a cross.*
Attribute:	**line-thickness**	**Description**: *Sets shape line-width, as measured in printable dots.*
Name/Value Pair:	line-thickness=*"4"*	
Units:	*dots*	**Options**: *1-100 dots.*
Type:	float	
Attribute:	**fill**	**Description**: *Sets shape background filling/shading.*
Name/Value Pair:	fill=*"none"*	
Values:	*none/solid*	**Options**: *none/solid*
Type:	string	

Element:	**<line>**	**Description**: *Child of <label> element. Creates a line.*
Attribute:	**start-x**	**Description**: *Required. Sets start position (horizontal displacement) of the line as measured from home position.*
Name/Value Pair:	start-x=*".03"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**start-y**	**Description**: *Required. Sets start position (vertical displacement) of the line, as measured from home position.*
Name/Value Pair:	start-y=*".06"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**end-x**	**Description**: *Required. Sets end position (vertical distance) of the line, as measured from home position.*
Name/Value Pair:	end-x=*"4"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**end-y**	**Description**: *Required. Sets end position (horizontal distance) of the line, as measured from home position.*
Name/Value Pair:	end-y=*".25"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**line-thickness**	**Description**: *Sets the line-width, as measured in printable dots.*
Name/Value Pair:	line-thickness=*"4"*	
Units:	*dots*	**Options**: *1-100 dots*
Type:	float	

Element:	**<arrow>**	**Description**: *Child of <label> element. See "BPL Glossary" for a description of the arrow-head parts. Creates an arrow.* ⇨.
Attribute:	**position-x**	**Description**: *Required. Sets position (horizontal displacement) for the start of the (invisible) box in which the arrow (including arrow head) is contained, as measured from the home position.*
Name/Value Pair:	position-x=*".03"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**position-y**	**Description**: *Required. Sets position (vertical displacement) for the start of the (invisible) box in which the arrow (including arrow head) is contained, as measured from the home position.*
Name/Value Pair:	position-y=*".06"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**height**	**Description**: *Required. Sets arrow height (vertical distance), as measured from starting position of the arrow. Includes the stem and arrow head.*
Name/Value Pair:	height=*".5"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**width**	**Description**: *Required. Sets arrow width (horizontal distance), as measured from starting position of the arrow.*
Name/Value Pair:	width=*"1"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	

Element:	**<arrow>** *(cont'd)*	**Description**: *Child of <label> element. See "BPL Glossary" for a description of the arrow-head parts. Creates an arrow.* .
Attribute:	**type**	**Description**: *Specifies type of arrow.*
Name/Value Pair:	type=*"right-arrow"*	
Units:	*(see Options list)*	**Options**: *right-arrow (* *)* *left-arrow (* *)* *up-arrow (* *)* *down-arrow (* *)* *rotational-arrow-clockwise (* *)* *rotational-arrow-counter-clockwise (* *)* *rotational-arrow-both-directions (* *)* *angled-arrow-right (* *)* *angled-arrow-left (* *)*
Type:	string	
Attribute:	**rotation**	**Description**: *Sets amount of clockwise rotation (number of degrees) to apply to the arrow.*
Name/Value Pair:	rotation=*"0"*	
Units:	*deg*	**Options**: *0-359*
Type:	integer	
Attribute:	**line-thickness**	**Description**: *Sets arrow line-width, as measured in printable dots.*
Name/Value Pair:	line-thickness=*"4"*	
Units:	*dots*	**Options**: *1-100 dots.*
Type:	float	

Element:	**<arrow>** *(cont'd)*	**Description**: *Child of <label> element. See "BPL Glossary" for a description of the arrow-head parts. Creates an arrow.* ⇨.
Attribute:	**fill**	**Description**: *Sets the arrow background filling/shading.*
Name/Value Pair:	fill=*"none"*	
Values:	*none/solid*	**Options**: *none/solid*
Type:	string	
Attribute:	**arrow-head-length**	**Description**: *Sets the length of the arrow head for rotational and angled arrows.*
Name/Value Pair:	arrow-head-length=*"3"*	
Values:	*headLength*	**Options**: *1-8.*
Type:	integer	
Attribute:	**arrow-head-right**	**Description**: *Sets the arrow angle direction to the right for angled arrows.*
Name/Value Pair:	arrow-head-right=*"1"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	integer	
Attribute:	**arrow-head-left**	**Description**: *Sets the arrow angle direction to the left for angled arrows.*
Name/Value Pair:	arrow-head-left=*"1"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	integer	

Element:	**<text>**	**Description**: *Child of <label> element. Creates a text object.*
Attribute:	**position-x**	**Description**: *Required. Sets position (horizontal displacement), as measured from home position.*
Name/Value Pair:	position-x=*".03"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**position-y**	**Description**: *Required. Sets position (vertical displacement), as measured from home position.*
Name/Value Pair:	position-y=*".06"*	
Units:	*inches/mm/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**rotation**	**Description**: *Sets amount of clockwise rotation (number of degrees) to apply to the text.*
Name/Value Pair:	rotation=*"45"*	
Units:	*deg*	**Options**: *0-359*
Type:	integer	
Attribute:	**align**	**Description**: *Specifies how the text should be aligned in the text object. This property has no effect on free-form sized text objects.*
Name/Value Pair:	align=*"center"*	
Values:	*left/right/center*	**Options**: *left, right, center.*
Type:	string	
Attribute:	**font-name**	**Description**: *Sets the font family that will be used for the text.*
		Note: Fonts are printer specific. See your printer manual for a list of printer-specific options.
Name/Value Pair:	font-name=*"Arial"*	
Values:	*fontName*	**Options**: *fontName*
		Note: If the font name is misspelled, the text will be printed with the printer-specific default font.
Type:	string	

Element:	**<text>** *(cont'd)*	**Description**: *Child of <label> element. Creates a text object.*
Attribute:	**italic**	**Description**: *Specifies whether the text will be italicized.*
Name/Value Pair:	italic=*"false"*	
Values:	*true/false*	**Options**: *true/false.*
Type:	boolean	
Attribute:	**bold**	**Description**: *Specifies whether the text will be bolded.*
Name/Value Pair:	bold=*"false"*	
Values:	*true/false*	**Options**: *true/false*
Type:	boolean	
Attribute:	**underline**	**Description**: *Specifies whether the text will be underlined.*
Name/Value Pair:	underline=*"false"*	
Values:	*true/false*	**Options**: *true/false*
Type:	boolean	
Attribute:	**vertical-text**	**Description**: *Specifies whether the text will be oriented vertically.*
Name/Value Pair:	vertical-text=*"true"*	
Values:	*true/false*	**Options**: *true/false*
Type:	boolean	
Attribute:	**reverse-text**	**Description**: *Specifies whether the text will be reverse oriented.*
Name/Value Pair:	reverse-text=*"false"*	
Values:	*true/false*	**Options**: *true/false*
Type:	boolean	
Attribute:	**mirror-text**	**Description**: *Specifies whether text will be set to mirror image.*
Name/Value Pair:	mirror-text=*"false"*	
Values:	*true/false*	**Options**: *true/false*
Type:	boolean	

Element:	**<text>** *(cont'd)*	**Description**: *Child of <label> element. Creates a text object.*
Attribute:	**show-bounding-box**	**Description**: *Specifies if there is a bounding box around text.*
Name/Value Pair:	show-bounding-box=*"true"*	
Values:	*true/false*	**Options**: *true/false*
Type:	boolean	
Attribute:	**line-thickness**	**Description**: *Specifies line thickness of the bounding box. Has no effect if a bounding box is not used.*
Name/Value Pair:	line-thickness=*"4"*	
Units:	*dots*	**Options**: *dots*
Type:	integer	

Element:	<text-sizing>	**Description**: *Must have a start-text-sizing and end- text-sizing tag around the text-sizing options. Child of <text> element. Specifies how the size of the text object will be set or calculated.*
Name/Value Pair:	<text-sizing>	**Description**: *Starts the text-sizing element.*
	</text-sizing>	**Description**: *Ends the text-sizing element.*
Child Element:	<autosize>	**Description**: *Child of <text-sizing> element. User specifies the box height and box width. As text is entered, the autosize command recalculates the font size so the text will print at the largest size possible and still fit into the defined height and width (text will wrap).*
Attribute:	**height**	**Description**: *Required. Sets height of the text object.*
Name/Value Pair:	height=*"3"*	
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**width**	**Description**: *Required. Sets width of the text object.*
Name/Value Pair:	width=*"4"*	
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Child Element:	<free-form>	**Description**: *Child of <text-sizing> element. User sets the height of the text box. There is no limit to the amount of text entered (text will wrap), however, if the text exceeds the size of the label, it will run off the label and not be printed.*
Attribute:	**height**	**Description**: *Required. Sets height of the text object.*
Name/Value Pair:	height=*"3"*	
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	

Element:	<text-sizing> *(cont'd)*	**Description**: *Must have a start-text-sizing and end- text-sizing tag around the text-sizing options. Child of <text> element. Specifies how the size of the text object will be set or calculated.*
Child Element:	<manual>	**Description**: *Child of <text-sizing> element. User sets the size of the text box and the font size used. If the text exceeds the size of the box (text will wrap), the end will not be printed.*
Attribute:	**height**	**Description**: *Required. Sets height of the text object.*
Name/Value Pair:	height=*"3"*	
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**width**	**Description**: *Required. Sets width of the text object.*
Name/Value Pair:	width=*"6"*	
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**font-size**	**Description**: *Required. Sets font size to be used for the text object.*
Name/Value Pair:	font-size=*"12"*	
Values:	*points*	**Options**: *points (must be positive)*
Type:	float	

Element:	<datasource>	**Description**: *Defines how data will be retrieved (i.e., static or dynamic - from keyboard, RTC, barcode, scanner or serialization).*
Syntax:	<datasource>	**Description**: *Child of <text> or <barcode> element. Starts a datasource element. Datasource elements are used for static text in both text and barcode elements. Must have a start- and end- datasource element.*
	</datasource>	**Description**: *Ends the datasource element.*
Child Element:	<static-text>	
Attribute:	**value**	**Description**: *Required. Specifies text string that will be displayed.*
Name/Value Pair:	value=*"Hello"*	
Values:	*string*	**Options**: *string* **Note:** To enter double quotes (") in the text, use either: - single quote on the value (e.g., <value='Test "Data"'/>) - or - the " notation (e.g., value="Test "Data"").
Type:	string	
Child Element:	<prompt-text>	
Attribute:	**prompt**	**Description**: *Required. Specifies the prompt the user will be shown at print time to retrieve the data for this object.* **Note:** If this option is selected, each time the label is printed, the printer will stop and prompt the user for input. The label will not be printed until all prompted text is entered.
Name/Value Pair:	prompt=*"Enter first name"*	
Values:	*promptString*	**Options**: *promptString*
Type:	string	
Attribute:	**default**	**Description**: *Required. Specifies the default text placed inside a text box. Can be blank or any message/question desired.* **Note:** The text box is used to accept user responses to a prompt message.
Name/Value Pair:	default=*"Mary"*	
Values:	*Text*	**Options**: *Text*
Type:	string	

Element:	<datasource> *(cont'd)*	**Description**: *Defines how data will be retrieved (i.e., static or dynamic - from keyboard, RTC, barcode, scanner or serialization).*
Child Element:	<date-time>	
Attribute:	**date-time-format**	**Description**: *Required (see note). Queries current date from printer and sets printer date format for insertion on the label.*
		Note: At least one of the date/time formats is required (date-format or time-format). You can also use both.
Name/Value Pair:	date-time-format="2"	
Values:	*2*	**Options**:

0	*M/d/yyyy*	*(3/25/2011)*
1	*M/d/yy*	*(3/25/11)*
2	*MM/dd/yy*	*(03/25/11)*
3	*MM/dd/yyyy*	*(03/25/2011)*
4	*yy/MM/dd*	*(11/03/25)*
5	*yyyy-MM-dd*	*(2011-03-25)*
6	*d-MMM-yy*	*(25-Mar-11)*
7	*dddd, MMMM d, yyyy*	*(Friday, March 25, 2011)*
8	*MMMM d, yyyy*	*(March 25, 2011)*
9	*dddd, d MMMM, yyyy*	*(Friday, 25 March, 2011)*
10	*d MMMM, yyyy*	*(25 March, 2011)*
11	*h:mm:ss tt*	*(8:55:31 AM)*
12	*hh:mm:ss tt*	*(08:55:31 AM)*
13	*H:mm:ss*	*(8:55:31)*
14	*HH:mm:ss*	*(8:55:31)*
15	*MM/dd/yy h:mm tt*	*(03/25/11 8:55 AM)*
16	*MM.dd.yy h:mm tt*	*(03.25.11 8:55 AM)*
17	*dd/MM/yy h:mm tt*	*(25/03/11 8:55 AM)*
18	*dd.MM.yy hh:mm tt*	*(25.03.11 08:55 AM)*
19	*hh:mm tt*	*(08:55 AM)*

Type:	integer

Element:	<datasource> *(cont'd)*	**Description***: Defines how data will be retrieved (i.e., static or dynamic - from keyboard, RTC, barcode, scanner or serialization).*
Element:	<sequence>	
Attribute:	**start**	**Description***: Required. Sets starting value.*
Name/Value Pair:	start=*"1"*	
Values:	*StartNum*	**Options**: *StartNum*
Type:	integer	
Attribute:	**increment**	**Description***: Required. Sets incremental value.*
Name/Value Pair:	increment=*"1"*	
Values:	*Incr*	**Options**: *Incr*
Type:	integer	
Attribute:	**number-of-labels**	**Description***: Required. Sets total number of labels to print.*
Name/Value Pair:	number-of-labels=*"10"*	**Note:** If the label includes more than one serial object, the result is one complete set of the labels generated from the serial object that prints the *most* labels; the shorter series will repeat until the longer series is completed.
Units:	*NumLabels*	**Options**: *NumLabels*
Type:	integer	
Attribute:	**prefix**	**Description***: Required. Adds a constant prefix string to the sequence data.*
Name/Value Pair:	prefix=*"P/N"*	
Values:	*pre*	**Options**: *pre*
Type:	string	
Attribute:	**postfix**	**Description***: Required. Adds a constant string after the sequence data.*
Name/Value Pair:	postfix=*"Rev A"*	
Values:	*post*	**Options**: *post*
Type:	string	

Element:	**<barcode>**	**Description:** *Creates a barcode object on the label.*
Attribute:	**position-x**	**Description**: *Required. Sets position (horizontal displacement), as measured from the home position.*
Name/Value Pair:	position-x=*".03"*	
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**position-y**	**Description**: *Required. Sets position (vertical displacement), as measured from the home position.*
Name/Value Pair:	position-y=*".06"*	
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**height**	**Description**: *Required. Sets barcode height.*
Name/Value Pair:	height=*"2"*	
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**type**	**Description**: *Required. Sets barcode type.*
Name/Value Pair:	type=*"code 39"*	
Values:	*typename*	**Options** *(must be typed exactly as shown): code 39, code 93, code 128 a, code 128 b, code 128 c, codabar a/f, ean 8, ean 13, ean 128, ean/ucc 128, ean/upc ext2, ean/upc ext5, fim, hibc, interleaved 2 of 5, jan 8, jan 13, msi plessy, upc a, upc e, upc e ext2, upc e ext5, code 16k, postnet, aztec, datamatrix, pdf 417, micro pdf 417, ups maxicode, qr-code, code 49* **Note:** Your printer may support a different barcode set. See your printer manual for a list of printer-specific options.
Type:	string	
Attribute:	**rotation**	**Description**: *Sets the amount of clockwise rotation (number of degrees) to apply to the barcode.*
Name/Value Pair:	rotation=*"0"*	
Units:	*deg*	**Options**: *0-359*
Type:	integer	

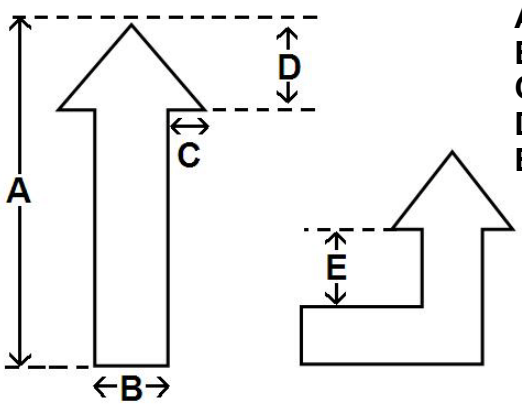
Element:	**<barcode>** *(cont'd)*	**Description:** *Creates a barcode object on the label.*
Attribute:	**human-readable**	**Description**: *Sets whether human-readable text will be included on barcode.*
Name/Value Pair:	human-readable=*"true"*	
Values:	*true/false*	**Options**: *true, false*
Type:	boolean	
Attribute:	**human-readable-location**	**Description**: *Sets location of human-readable text string on the barcode. Takes effect only if human-readable is "true."*
Name/Value Pair:	human-readable-location= *"bottom"*	
Values:	*top/bottom*	**Options**: *top, bottom.*
Type:	string	
Attribute:	**density**	**Description**: *Sets the print density for the barcode.*
Name/Value Pair:	density=*"20"*	
Values:	*density*	**Options**: *10, 20, 30, 40, 50, 60, 70, 80, 90*
Type:	integer	
Attribute:	**ratio**	**Description**: *Sets the thickness (overall width) of the barcode. Does not apply to all symbologies.*
Name/Value Pair:	ratio=*"2:1"*	
Values:	*ratio*	**Options**: *2:1, 2.5:1, 3:1*
Type:	integer	
Attribute:	**check-character**	**Description**: *Sets whether a check character will be included on barcode. If true, the human-readable text will also include the check character.*
Name/Value Pair:	check-character=*"true"*	
Values:	*true/false*	**Options**: *true, false*
Type:	boolean	

Element:	**<graphic>**	**Description**: *Creates a graphic object on the label.*
Attribute:	**position-x**	**Description**: *Required. Sets position (horizontal displacement), as measured from the home position.*
Name/Value Pair:	position-x=*".03"*	
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**position-y**	**Description**: *Required. Sets position (vertical displacement), as measured from the home position.*
Name/Value Pair:	position-y=*".06"*	
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**height**	**Description**: *Required. Sets graphic height.*
Name/Value Pair:	height=*"2"*	
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	float	
Attribute:	**width**	**Description**: *Required. Sets graphic width.*
Name/Value Pair:	type=*"4"*	
Units:	*inches/millimeters/dots*	**Options**: *inches, millimeters, dots*
Type:	string	
Attribute:	**rotation**	**Description**: *Sets the amount of clockwise rotation (number of degrees) to apply to the barcode.*
Name/Value Pair:	rotation=*"0"*	
Units:	*deg*	**Options**: *0-359*
Type:	integer	
Attribute:	**file-name**	**Description**: *Specifies the name of the graphic file that will be used.*
Name/Value Pair:	file-name= *"BradyLogo.jpg"*	
Values:	*fileName*	**Options**: *fileName*
Type:	string	

A Glossary

BPL Glossary

- **Arrow-head Definitions:**



A = arrow height
B = arrow-head-length
C = arrow-head-overhang
D = arrow-stem-width
E = base-arrow-head-length

- **Attributes**: Appears within the opening tag of an element (see Elements definition) and are used to include additional information about the element. If an attribute has multiple units, the value of the attribute must be specified (e.g., if a tag has a width attribute, the units would be: inches, mm, dots).

  Example: *<star position-x="" position-y="" height="2" width="2" />*

  Element: *star*
  Attributes: *positon-x, position-y, height, width*
  Value: *"value"*

- **Child Element**: Follow the same format as the root element. They must have matching opening and closing tags. Child elements are exactly one level lower than a parent element. You may add as many child elements to the root as necessary.

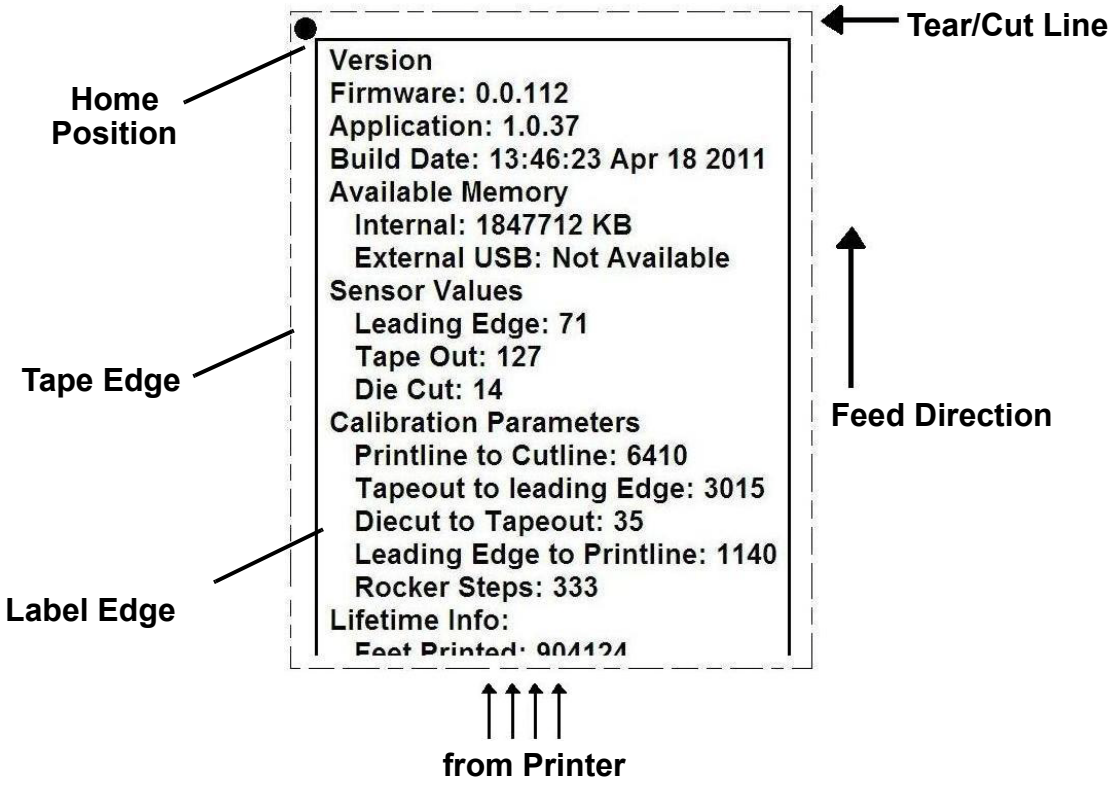
  Example:

  root/parent: *<labels>*
  parent: *<label>*
  child: *<rectangle />*
  child: *<circle />*
  parent: *</lable>*
  root/parent: *</labels>*

- **Container**: In XML, a container (also called a wrapper element) is an element used to organise or group elements in the XML file. The first container is always the Root Element, with other containers used to enclose a group of similar elements (e.g., a label container can contain a group of elements such as a rectangle, circle, start, etc.).
- **Character Set**: BPL is compatible with ANSI and Unicode UTF-8 character sets.
- **Comma Separators**: Not valid in any place except between units in a string.
- **Comments**: Define or explain a coding section. Comments can be placed anywhere in the script, however, they cannot be placed on the same line as an element.

  All text between <!- - and - - > is ignored. Comments can span more than one line, as long as the sequence of 'end comment' characters (-->) have not been entered. To insert a forced line break in a comment, just press Return on the keyboard.
- **Declaration Statement**: Defines the language and version and is the first line of any XML document.
  Example: *<?xml version="1.0" ?>*
- **Element**: Identify the operation(s) that will produce a desired outcome and are the basic building blocks of the file (similar to the "sentences" or "paragraphs" of a document). An XML file consists of a series of elements (nested and sequential). Each element consists of a start tag, one or more elements or attributes and a closing tag.
  Example:
- **Start Tag**: <defaults>
  Attributes: <label width="100" height="200" />
  <>printer match-media="12345" />
  End Tag: </defaults>
- **Element Format:** All BPL elements consist of a keyword followed by zero or more attributes. Elements are case sensitive.
- **Element Tags**: Establish a hierarchical syntax. Names should be recognizable and easily managed (e.g., for a label, use key names such as 'barcode' and within 'barcode', list more detailed tags, such as 'code 39' or 'ean 128'). XML does not use static tags. All element tags must have closing tags. Element tags are case-sensitive; beginning and end tags must match exactly.
  Example: *<label> - opening tag, </barcode> - closing tag*
- **Extensible Markup Language (XML)**: A set of rules for encoding documents in machine-readable form. Emphasizes simplicity, generality, and usability.

- **Hierarchy**: Good structure and hierarchy consists of begin-, end-, and empty-element tags which delimit the elements are correctly nested, with none missing and none overlapping.
- **Home Position**: Upper-left corner of label.



- **Markup and Content** : The characters in XML documents are divided into *markup* and *content*, which are distinguished by syntactic rules. All markup strings either begin with the character "<" and end with a ">", or begin with the character "&" and end with a ";". All other strings of characters are content.

- **Namespaces:** A "prefix" used to differentiate attributes/elements in an XML file. Namespaces can be declared on any element to define a namespace local to that element and/or any children of that element.

  The current BPL namespace is **http://www.bradycorp.com/printers/bpl** (e.g <bpl-document xmlns="http://www.bradycorp.com/printers/bpl">).
- **Parent Element**: Can contain one or more Child elements. Parent elements must have start "<" and end ">" tags, or be an empty-element tag.
- **Quote Characters**: All units in BPL are placed inside of quotation marks. If your string text uses quotes (") in the text, use one of these options:
  - Single quote the attribute unit (e.g., <unit='Test "Data"' />)
  - Use the " notation (e.g., value=""Test" Text").
- **Root Element**: A single "root" element which is the container for all other elements (e.g., start tag = <bpl-document>, end tag = </bpl-document>).
- **Spaces:** If an element has more than one attribute, spaces (not commas) are used between attribute/value pairs.
- **Start-End Characters:** All elements must must have a start- and end- tag. You can end an element with /> or use an end tag (</endElement>) as you would if there were other elements or attributes between them.
- **Tag**: A generic name for an element. All information that belongs to an element must be contained between the starting "<" and ending ">" tags of an element. There are three types: *start-tags (e.g.,* `<section>`), *end-tags (e.g.,* `</section>`), and *empty-element tags (e.g.,* `<line-break />`).
- **Termination Characters**: All elements require a termination tag. You can also end an element by typing the termination character at the end of the element (e.g., <defaults />).
- **Type**: Identifies the string type.
- **Unicode**: XML supports almost any Unicode character in element names, attributes, comments, character data, and processing instructions (other than the ones that have special symbolic meaning in XML itself (e.g., less-than sign, "<"). Almost every legal Unicode character may appear in an XML document.
- **Whitespace**: Can be used anywhere in an XML file (except within quoted units) to improve readability.

© 2013 Brady Worldwide, Inc. All Rights Reserved.