CHARACTER BITMAP DISTRIBUTION FORMAT Specification Version 2.1

January 16, 1989
PostScript® Developer Tools & Strategies Group

Adobe Systems Incorporated
1585 Charleston Road PO Box 7900
Mountain View, CA 94039-7900
(415) 961-4400

PN LPS5005

Copyright © 1989, 1988, 1987 by Adobe Systems Incorporated.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher.

PostScript is a registered trademark of and the PostScript logo is a trademark of Adobe Systems Incorporated.

The information herein 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 this book. The software described in this book is furnished under license and may only be used or copied in accordance with the terms of such license.

CHARACTER BITMAP DISTRIBUTION FORMAT Specification Version 2.1

January 16, 1989
Adobe Systems Incorporated

1. INTRODUCTION

This document describes Adobe Systems's character bitmap distribution format. The format is intended to be easily understood by both humans and computers. The format described in this document is subject to change without prior notification.

1.1 TAPE FORMAT

These character bitmaps are typically distributed on magnetic tape. Each tape is 1600 BPI, nine track, unlabeled, and contains two or more files. Each file is followed by an EOF mark. The last file on the tape is followed by two EOF marks. Physical records contain 512 bytes. The last physical record in a file (preceding an EOF mark) may contain fewer than 512 bytes.

Each file is encoded in the printable characters (octal 40 through 176) of USASCII plus carriage return and linefeed. Each file consists of a sequence of variable-length lines. Each line is terminated by a carriage-return (octal 015) and line-feed (octal 012). The first file on the tape is the Adobe Systems Copyright notice. Following files are font files. The format of font files is described in the following sections.

Note:
Font tapes may also be obtained in UNIX **tar** *format. Be sure to specify tar format if desired. No other tape formats are currently supported by Adobe Systems.*

1.2 FILE FORMAT

Character bitmap information is distributed in an USASCII encoded, human readable form. The information about a particular family and face at one size and orientation is contained in one file. The file begins with information pertaining to the face as a whole, followed by the information and bitmaps for the individual characters.

A font bitmap description file has the following general form, where each item is contained on a separate line of text in the file. Items on a line are separated by spaces.

GLOBAL HEADER INFORMATION

- The word STARTFONT followed by a version number indicating the exact file format used (for example, 2.1)
- One or more lines beginning with the word COMMENT. These lines may be ignored by any program reading the file.

©1988 Adobe Systems Incorporated. All rights reserved.

- The word FONT followed by the **family name** and the **face name** separated by a hyphen. This should exactly match the PostScript outline font name.

- The word SIZE followed by the **point size** of the characters, the **x resolution**, and the **y resolution** of the device for which these characters were intended. All are represented as integers.

- The word FONTBOUNDINGBOX followed by the **width in x**, **height in y**, and the **x and y displacement** of the lower left corner from the **origin**. (See the examples in section 1.3). These are all integers.

- Optionally the word STARTPROPERTIES followed by the number of properties (**p**) that follow. This is a recent addition to the format. Within the properties list, there may be **p** lines consisting of a word for the **property name** followed by either an integer or string surrounded by ASCII double quotes (ASCII octal 042). Internal quote characters are indicated (or "quoted") by using two in a row. The property section, if it exists, is terminated by ENDPROPERTIES

THE INDIVIDUAL CHARACTER INFORMATION

The character section is introduced by the word CHARS followed by the number of character segments (**c**) that follow. This is an integer value. Error checking is recommended at the end of the file, to make sure that **c** characters were actually read and processed. Each of the **c** characters is then represented by the following:

- The word STARTCHAR followed by up to 14 bytes (no blanks) containing the **name** of the glyph. This should correspond to its name in the PostScript outline font's encoding vector.

- The word ENCODING followed by a positive integer representing the Adobe Standard Encoding value. If the character is *not* a member of the Adobe Standard Encoding, ENCODING is followed by -1 and optionallly by another integer specifying the glyph index.

- The word SWIDTH followed by the **scalable width** in x and y of character. Scalable widths are in units of 1/1000th of the size of the character, and correspond to the widths found in AFM files (for outline fonts). If the size of the character is *p* points, the width information must be scaled by *p*/1000 to get the width of the character in printer's points. This width information should be considered as a vector indication the position of the next character's origin relative to the origin of this character. To convert the scalable width to the width in device pixels, multiply SWIDTH times *p*/1000 times *r*/72 where *r* is the device resolution in pixels per inch. The result is a real number giving the ideal print width in device pixels. The actual device width must of course be an integral number of device pixels and is given in the next entry.

- The word DWIDTH followed by the width in x and y of the character in device units (pixels). Like the SWIDTH, this width information is a vector indicating the position of the next character's origin relative to the origin of this character.

- The word BBX followed by the **width** in x (*BBw*), **height** in y (*BBh*) and x and y displacement (*BBxoff, BByoff*) of the lower left corner of the bitmap from the **origin** of the character.

- The word BITMAP. This introduces the hexadecimal data for the character bitmap.

©1988 Adobe Systems Incorporated. All rights reserved.

- From the BBX value for *h*, find *h* lines of **hex-encoded bitmap**, padded on the right with zero's to the nearest byte (i.e., multiple of 8). Hex data can be turned into binary by taking two bytes at a time, each of which represents 4 bits of the 8-bit value. For example, the byte 01101101 is two hex digits: 6 (0110 in hex) and D (1101 in hex).

- The word ENDCHAR.

- The entire file is terminated with the word ENDFONT. If this is encountered before **c** characters have been read, it is an error condition.

1.3 METRIC INFORMATION

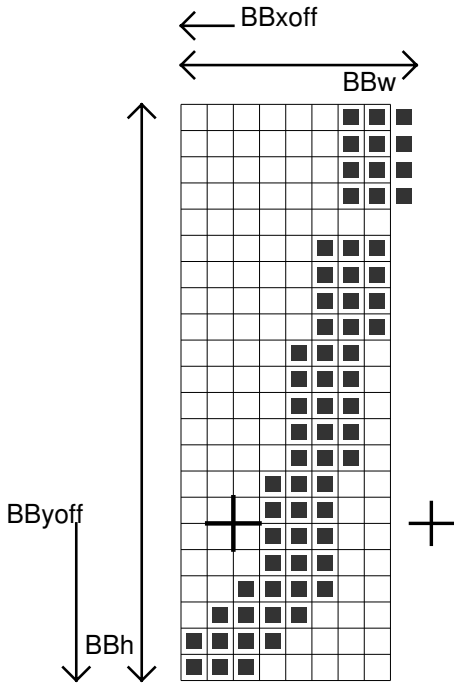
The font metrics include both the scalable width (really the width of the corresponding printer font character) and the character width of the screen font glyph, expressed in pixels. The scalable width is more accurate, and can be used by applications for keeping track of roundoff error and compensating in placement.

The following figures best illustrate the bitmap format and character metric information:

```
STARTCHAR j
ENCODING 106
SWIDTH 355 0
DWIDTH 8 0
BBX 9 22 -2 -6
```

The Bounding Box is expressed differently than other PostScript language files; the first two are the *width* and *height*, the second two are the offsets in *x* and *y*. This can bee seen in the illustration at right.

The character width from the origin (between + indicators) is 8 pixels, which has nothing to do with the actual bits, but is how far the current point moves after rendering the character.

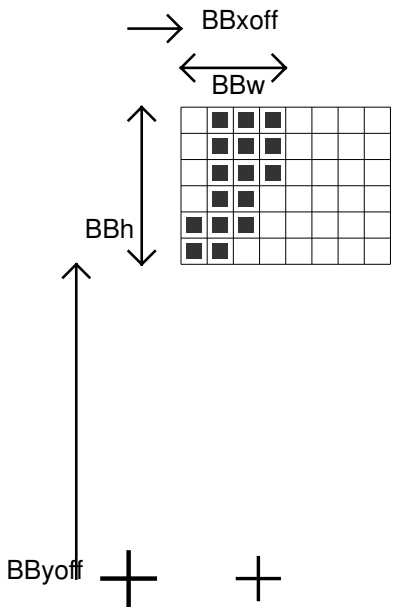


The bounding box of the bitmap character can be used to predict how much data to read in the BITMAP section; the first two numbers give the width and height of the bitmap, and correspond exactly to how much data is supplied. The offset then allows positioning without repeating lots of white bits (look at the following **quoteright** character, which doesn't have very many bits, but is located far above the baseline. That is what the offset fields are for):

©1988 Adobe Systems Incorporated. All rights reserved.

```
STARTCHAR quoteright
ENCODING 39
SWIDTH 223 0
DWIDTH 5 0
BBX 4 5 2 12
```

Here the actual bitmap is much smaller, and the offset (2 in *x*, 12 in *y*) positions the glyph with respect to its origin. These bitmaps are actually both from an italic font; notice that the character width of the quoteright leaves the origin still to the *left* of the actual bits after the character is drawn. Since all the characters are slanted, the next one will not interfere.



The bitmap itself is started by the BITMAP keyword and finished with the ENDCHAR keyword. It is best to "predict" the amount of data needed (using the BBX information) and use the ENDCHAR as an error-checking method: if you have consumed what you think is the appropriate amount of data, the very next thing in the file should be ENDCHAR. If not, either your parser is in error or the file is not complete (or is incorrect).

The bitmap itself is represented as hexadecimal digits, where each row corresponds to one row of the character bitmap. The bits are padded out to the nearest byte boundary with 0's, and the BBX bounding box information should be carefully consulted to determine how to extract the data.

©1988 Adobe Systems Incorporated. All rights reserved.

The following is an abbreviated example of a bitmap file containing the specification of two characters (the j and quoteright from the previous examples):

```
STARTFONT 2.1
COMMENT This is a sample font in 2.1 format.
FONT Helvetica-BoldOblique
SIZE 8 200 200
FONTBOUNDINGBOX 9 24 -2 -6
STARTPROPERTIES 2
MinSpace 4
Copyright "Copyright (c) 1987 Adobe Systems, Inc."
ENDPROPERTIES
CHARS 2
STARTCHAR j
ENCODING 106
SWIDTH 355 0
DWIDTH 8 0
BBX 9 22 -2 -6
BITMAP
0380
0380
0380
0380
0000
0700
0700
0700
0700
0E00
0E00
0E00
0E00
0E00
1C00
1C00
1C00
1C00
2C00
7800
F000
E000
ENDCHAR
STARTCHAR quoteright
ENCODING 39
SWIDTH 223 0
DWIDTH 5 0
BBX 4 5 2 12
BITMAP
70
70
60
E0
C0
ENDCHAR
ENDFONT
```

©1988 Adobe Systems Incorporated. All rights reserved.