Level 2 Compatibility: The setscreen and currentscreen Operators

Adobe Developer Support

Technical Note #5119

31 March 1992

Adobe Systems Incorporated
Adobe Developer Technologies
345 Park Avenue
San Jose, CA 95110
http://partners.adobe.com/
Contents

Level 2 Compatibility: The setscreen and currentscreen Operators 5

1 Introduction 5

2 Operator Behavior 5
   Level 1 Behavior 5
   Level 2 Behavior 6

3 Compatibility Implications 7

Appendix: Changes Since Earlier Versions 9
Level 2 Compatibility: The setscreen and currentscreen Operators

1 Introduction

With the introduction of PostScript™ Level 2, control over halftoning in the PostScript interpreter has been greatly improved. Because of this, the currentscreen and setscreen operators have been extended to accept two sets of parameters.

This technical note documents a modification made to the behavior of currentscreen and setscreen, which was changed to increase the backward compatibility of Level 2. This document corrects early versions of the PostScript Language Reference Manual, Second Edition, and addresses possible compatibility conflicts between Level 1 and Level 2.

2 Operator Behavior

The setscreen and currentscreen operators behave differently in Level 1 and Level 2 implementations due to the expanded functionality in Level 2.

2.1 Level 1 Behavior

Halftoning in Level 1 interpreters is controlled by using the setscreen operator, which allows an application to specify a halftone screen by passing the following parameters:

\[ \text{frequency angle proc} \]

Similarly, the currentscreen operator queries the current screen and returns

\[ \text{frequency angle proc} \]

on the stack. In both cases, the proc is a PostScript language procedure responsible for determining the order in which pixels in a halftone cell are turned on. (Refer to section 6.4, “Halftones,” in the PostScript Language Reference Manual, Second Edition.)
2.2 Level 2 Behavior

The concept of a halftone dictionary was introduced in Display PostScript™ systems to support greater language-level control over halftoning. This functionality is also a standard in PostScript Level 2. A halftone dictionary is a dictionary object whose entries are parameters to the halftoning machinery. When the PostScript interpreter starts, a current halftone dictionary exists, much like a current screen in Level 1 interpreters.

Halftone dictionaries can be passed to the sethalftone operator, which reads the relevant keys and sets up the halftone screen appropriately. There are several halftone dictionary types that are specified with the required HalftoneType key.

A dictionary of halftone type 1 consists of required entries similar to the values passed to the setscreen operator in PostScript Level 1, but this dictionary may also contain several optional entries that provide greater control over the halftone screen. Although it is possible to use setscreen to set up a screen based on a halftone dictionary, it is preferable to use sethalftone. (See section 3, “Compatibility Implications.”)

PostScript Level 2 extends the functionality of the currentscreen and setscreen operators to provide compatibility with the existing base of PostScript language code, while taking into account that the current halftone might be a dictionary rather than a spot function.

In PostScript Level 2 the current halftone can be defined either as a spot function or as a halftone dictionary. If it is a spot function, the currentscreen operator will behave the same way it would in Level 1, returning

```
frequency angle proc
```

on the operand stack. However, if the current halftone is a dictionary of type HalftoneType 1, rather than a screen, the currentscreen will return

```
frequency angle halftonedict
```

where frequency and angle are the values of Frequency and Angle extracted from the halftone dictionary. If the current halftone is a dictionary other than HalftoneType 1, currentscreen will return

```
60 0 halftonedict
```

where 60 and 0 are meaningless constants provided for language compatibility with existing PostScript language files. In similar fashion, the setscreen operator will accept both the Level 1 form for halftones

```
frequency angle proc
```
and the Level 2 form

\texttt{num1\ num2\ halftonedict}

If a halftone dictionary of type 1 is passed as the halftonedict, \texttt{setscreen} will interpret the arguments as

\texttt{frequency\ angle\ halftonedict}

\texttt{frequency} and \texttt{angle} will be copied into the halftone dictionary as \texttt{Frequency} and \texttt{Angle}. These values will override the \texttt{Frequency} and \texttt{Angle} keys in the halftone dictionary. If the dictionary is read-only, \texttt{setscreen} makes a copy of it before copying the values. If the halftone dictionary is a type other than 1, \texttt{frequency} and \texttt{angle} are ignored.

Some versions of the \textit{PostScript Language Reference Manual, Second Edition} state that the \texttt{frequency} and \texttt{angle} parameters returned by \texttt{currentscreen} are always 60 and 0 in the case where the current halftone is a dictionary. However, this is true only if the halftone dictionary is not type 1. Similarly, the manual states that \texttt{setscreen} always ignores the \texttt{frequency} and \texttt{angle} values passed on the stack if the halftone is a dictionary. Again, this is only true if the dictionary is not a type 1 halftone dictionary.

\textit{Note} The behavior described above is different from that described in some versions of the \textit{“PostScript Language Reference Manual, Second Edition.”}

3 Compatibility Implications

The ability to pass a halftone dictionary to the \texttt{setscreen} operator is provided for compatibility with existing PostScript language programs that execute \texttt{currentscreen}, save the results, and later execute \texttt{setscreen} to restore the old screen. It is preferable to use the \texttt{sethalftone} operator when explicitly setting a halftone dictionary as the current screen because it is more clear and because of the behavior described in the previous section.

Although the behavior of \texttt{setscreen} and \texttt{currentscreen} will be correct for most existing PostScript language files, there still remains a compatibility problem. In the case where the current halftone is not a type 1 dictionary, the \texttt{setscreen} and \texttt{currentscreen} operators behave as described in the \textit{PostScript Language Reference Manual, Second Edition}, that is, \texttt{currentscreen} will return the meaningless constants 60 and 0, and \texttt{setscreen} will ignore the first two values passed to it. Since the effect of the \texttt{setscreen} and \texttt{currentscreen} operators are device-dependent, the effect of returning or setting halftones with these constants is at least no worse than using these operators in the first place.
Appendix: Changes Since Earlier Versions

Changes since August 8, 1991 version

• Document was re-formatted in the new document layout and minor editorial changes were made.