Clipsave and Cliprestore

Adobe® Developers Association

10 October 1997

Technical Note #5607

LanguageLevel 3

Adobe Systems Incorporated

Corporate Headquarters
345 Park Avenue
San Jose, CA 95110-2704
(408) 536-6000

Eastern Regional Office
24 New England
Executive Park
Burlington, MA 01803
(617) 273-2120

Adobe Systems Europe Limited
Adobe House, Mid New Cultins
Edinburgh EH11 4DU
Scotland, United Kingdom
+44-131-453-2211

Adobe Systems Japan
Yebisu Garden Place Tower
4-20-3 Ebisu, Shibuya-ku
Tokyo 150 Japan
+81-3-5423-8100

PN LPS5607

Adobe Systems Incorporated

Contents

Adobe Systems Incorporated

Preface

This Document

This is the original release for *Clipsave and Cliprestore*, a document that provides a detailed description of new LanguageLevel 3 operators **clipsave** and **cliprestore**.

Intended Audience

This document is written for software developers who are interested in learning about the **clipsave** and **cliprestore** operators, or using these operators in an application that supports PostScript® printing devices.

It is assumed that the developer is already familiar with how the graphics state works in the PostScript language, including the operators **save**, **restore**, **gsave**, and **grestore**.

Organization of This Document

Section 1.1, “Overview of the Clipsave and ClipRestore Operators,” gives some general background of the graphics state and how the PostScript language has been extended with these two new operators to work explicitly with the current clipping region without effecting other parts of the current graphics state.

Section 1.2, “Implementing Clipsave and Cliprestore,” describes how the **clipsave** and **cliprestore** operators are defined in LanguageLevel 3. Included in this section is a very simple example that shows how these operators can be used to create smaller and more efficient code.

Related Publications

Supplement: PostScript Language Reference Manual (LanguageLevel 3 Specification and Adobe PostScript 3™ Version 3010 Product Supplement), available from the Adobe® Developers Association, describes the formal extensions to the PostScript language that have occurred since the publication of the PostScript Language Reference Manual, Second Edition. This supplement includes all LanguageLevel 3 extensions available in version 3010.

PostScript Language Reference Manual, Second Edition (Reading, MA: Addison-Wesley, 1991) is the developer's reference manual for the PostScript language. It describes the syntax and semantics of the language, the imaging model, and the effects of the graphical operators.

Statement of Liability

THIS PUBLICATION AND THE INFORMATION HEREIN IS FURNISHED AS IS, 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, MAKES NO WARRANTIES OF ANY KIND (EXPRESS, IMPLIED, OR STATUTORY) WITH RESPECT TO THIS PUBLICATION, AND EXPRESSLY DISCLAIMS ANY AND ALL WARRANTIES OF MERCHANTABILITY, FITNESS FOR PARTICULAR PURPOSES, AND NONINFRINGMENT OF THIRD-PARTY RIGHTS.

Clipsave and Cliprestore

1 Clipsave and Cliprestore

1.1 Overview of the Clipsave and ClipRestore Operators

In graphical device interfaces, a graphical object to be printed by the PostScript printer driver is delivered with an accompanying clip that may or may not be the same as the clip associated with the objects preceding it or following it.

With previous levels of the PostScript language, clipping regions in the PostScript printer driver were managed using the **gsave** and **grestore** operators. The side effect of this was that every time it was necessary to save or restore a clip region, a number of other graphics state parameters were also saved or restored, since **gsave** and **grestore** push or pop a copy of the entire current graphics state. This would include the current transformation matrix (CTM), the current path, the clip path, and the identity of the raster output device (usually a printer). For example, if the **grestore** operator was called to restore the clip region, then it might be necessary to set some of the other graphics state parameters such as the color space, the current color, the current font, and the current line width.

The LanguageLevel 3 **clipsave** and **cliprestore** operators save and restore, respectively, only the clipping region of the current graphics state. These operators enable clips to be established and removed without disturbing other graphics state parameters. This method results in the decrease in size of files generated by the PostScript printer driver.

Like the **gsave** and **grestore** operators, **clipsave** and **cliprestore** use an implicit stack. They are subordinate to **gsave** and **grestore** in exactly the same way that **gsave** and **grestore** are subordinate to the **save** and **restore** operators. That is, if the **grestore** operator is executed, all of the **clipsave** operations since the corresponding **gsave** operation are implicitly restored; if more **cliprestore** operations are executed than **clipsave** operations since the most recent **gsave**, the extra **cliprestore** operations only return the graphics state to the base state established by the most recent **gsave** operation.

Each graphics state has its own clip state stack, which will always contain at least one element, namely the current clip for that graphics state. When a graphics state is copied by the **setgstate** or **currentgstate** operators, the clip state associated with that graphics state is also copied, replacing the clip state stack of the destination graphics state.

1.2 Implementing Clipsave and Cliprestore

The **clipsave** operator pushes a copy of the current clipping path onto the clipping path stack. The saved state may be restored later by a matching **cliprestore** operator.

The clipping path is saved as part of the graphics state parameters when a **gsave** operation is performed; however, the **clipsave** operator saves only the clipping path without saving the other graphics state parameters.

The **cliprestore** operator resets the current clipping path from the one on the top of the clipping path stack and pops the clipping path stack, restoring the clipping path in effect at the time of the matching **clipsave** operation. This operator provides a way to restore only the clipping path without restoring all of the graphics state parameters associated with a **grestore** operation.

If there is no matching **clipsave** operation, or if the most recent **clipsave** operation preceded the more recent unmatched **gsave** operation, the **cliprestore** operator does not pop the clipping path stack, although it does restore the clipping path from the clipping path stack associated with the **gsave** operation.

The following two code samples show the effects of using the **clipsave** and **cliprestore** operators. In the first example, these two operators are not used. As a result, the **setfont** and **setcolor** operators have to be called for each string that is printed, even though the information does not change, since they are removed by the next **grestore** operator. In the second example, **clipsave** and **cliprestore** are used. The **setfont** and **setcolor** operators need only be called once since any future calls to **clipsave** or **cliprestore** do not effect them.

```
% Example 1: without clipsave and cliprestore
gsave
    167 254 562 62 rectclip
    F0S100 Setfont %Need to substitute this code
    0 0 0 setcolor
    204 204 moveto .4 0 (1200) ashow
grestore
gsave
    378 254 626 62 rectclip
    F0S100 Setfont %Need to substitute this code
    0 0 0 setcolor
    483 204 moveto .4 0 (1201) ashow
grestore
gsave
    657 254 347 62 rectclip
    F0S100 Setfont %Need to substitute this code
    0 0 0 setcolor
    762 204 moveto .4 0 (1202) ashow
grestore


% Example 2: with clipsave and cliprestore
clipsave
    167 254 562 62 rectclip
    F0S100 Setfont %Need to substitute this code
    0 0 0 setcolor
    204 204 moveto .4 0 (1200) ashow
cliprestore
clipsave
    378 254 626 62 rectclip
    483 204 moveto .4 0 (1201) ashow
cliprestore
clipsave
    657 254 347 62 rectclip
    762 204 moveto .4 0 (1202) ashow
cliprestore
```

Note *For more information on the **clipsave** and **cliprestore** operators, see Chapter 8 of the Supplement: PostScript Language Reference Manual.*

Adobe Systems Incorporated

Index