XFree86® and DPS

Juliusz Chroboczek, <jch@pps.jussieu.fr>

27 February 2001

1. Introduction

Display Postscript (or DPS for short) is a rendering extension for X11. DPS is slightly atypical in that it is based on *code mobility*, the ability to send executable code from client to server. Thus, a DPS client does not request that a line should be rendered; instead, it uploads code to the server which, when executed, causes a line to be drawn.

This document does not aim at teaching programming with DPS; as it is only a summary description of the DPS support code included with XFree86. More information about DPS, including a DPS bibliography, is available from the DPS extension site `<URL:http://dps.sourceforge.net>`.

> *Note: Adobe, PostScript and Display PostScript are trademarks of Adobe Systems Incorporated which may be registered in certain jurisdictions.*

As with all X11 extensions, DPS consists of client-side and server-side components. The DPS client side consists of a number of libraries and a number of tools for programming and configuration. The DPS server side may consist either of an X server extension, or of a client-side process known as the ''DPS agent.'' In this latter case, the term ''server-side'' is somewhat misleading.

At the time of writing, only the client side is included with XFree86; the server side must be obtained separately. Please see *server side* (section 3., page 2) later in this document for more information.

2. The DPS client side

The DPS client side consists of four libraries and a number of basic tools for programming and configuration; these are all included with XFree86.

2.1 Libraries

The `libpsres` library is a library for management of *PostScript resources*, on-disk files representing PostScript data structures such as fonts, font encodings, procsets, *etc.* It is closely related to the `makepsres` tool (see *client-side tools* (section 2.2, page 2) later in this document).

The basic DPS client library is `libdps`. This library contains a number of functions for connection establishment, resource management, as well as stubs for all standard PostScript operators. Normally, all DPS clients should link with `libdps`; in addition, `libdps` may be used for printing by non-DPS clients (this is done, for example, by Sun's JDK). This library is documented in [CLRM] and [CLSX].

The `libdpstk` library contains a number of additional utilities for managing DPS contexts, user paths and user objects, and for previewing EPS files. It is documented in [DPTX].

The libdpstkXm library contains four Motif widgets. The *DPS Scrolling Widget* is a DPS drawing area that automatically manages issues such as scrolling, scaling, client-side backing store, incremental redisplay, *etc.* The *Font Selection Box*, and its associated *Font Preview*, present a convenient and powerful interface for choosing scalable fonts. Finally, the *Color Picker* presents an interface for choosing colours using either of the RGB or HSV spaces. The latter three widgets are documented in [DPTX]; some summary Scrolling Widget documentation is available in the doc subdirectory of the DPS.tar.gz file, available from <URL:ftp://dps.sourceforge.net/pub/dps/DPS.tar.gz>.

The source code for libdpstkXm is included with XFree86; however, as it depends on Motif, this library is not built by default. A GTK-based library providing some of the functionality of libdpstkXm is available from the gtkDPS site <URL:http://www.gyve.org/gtkDPS/>.

2.1.1 Libdps and Xt

In X11R5, libdps did not depend on libXt. In X11R6, however, code was added to make the Xt main loop dispatch to sundry code on DPS events; with this addition, all programs that link with libdps need to link with libXt, whether they use Xt or not.

This state of affairs is unfortunately true of the version of libdps included with XFree86. We are currently considering various solutions to this problem (including the use of weak linker symbols or splitting off the guilty functions into a separate library).

2.2 Client-side tools

In addition to the libraries, the client side of DPS consists of two utilities.

The makepsres utility is used for managing PostScript resources. Its basic operation consists in walking recursively a filesystem tree, noting all resources, and then writing out a ''Unix PostScript Resources,'' file, basically a directory of all the resources found. This utility is documented in the makepsres(1) manual page.

The pswrap utility is a stub generator for PostScript clients. Roughly speaking, it takes as its input textual PostScript code, and generates a collection of C functions that transmit that code in pre-tokenised form to the DPS extension. Pswrap is documented in [PSWRAP].

2.3 Sample clients

XFree86 contains three sample DPS clients, dpsinfo, dpsexec and texteroids. They are documented in their respective manual pages.

A number of sample clients that depend on Motif are available in <URL:ftp://dps.sourceforge.net/pub/ftp/DPS.tar.gz>. Additional sample clients can be found as part of GtkDPS (see above).

The GNUstep environment can be compiled to use DPS for all rendering; for more information, please see the GNUstep site <URL:http://www.gnustep.org>.

3. The DPS server side

To use DPS clients, you need to install a DPS server side, which can be either a server extension (a ''DPS/X extension''), or a separate process (referred to, variously, either as a ''DPS/NX agent'' or, rather confusingly, as ''Client-Side DPS'' (CSDPS).

3.1 Display Ghostscript

Display Ghostscript (note the capitalisation), or DGS, is a client-side implementation of DPS based on the Ghostscript PostScript interpreter. DGS is still in beta at the time of writing; it does, however, provide a very usable implementation of DPS, although it still has some problems with the semantics of multiple DPS contexts.

DGS is available from the GNUstep download area <URL:http://www.gnustep.org/resources/sources.html>.

3.2 The DPS extension

The DPS extension is a much younger project aiming at producing an efficient server-side implementation of DPS. The extension is currently in a state best described as alpha; current versions are known to crash the X server under some circumstances.

The DPS extension is available from the DPS extension site <URL:http://dps.sourceforge.net>.

4. References

Links to electronic versions of all of these, and more, are available from the DPS extension site <URL:http://dps.sourceforge.net>.

[PLRM2] PostScript language reference manual. Adobe Systems, 2nd ed. Addison-Wesley, 1990. ISBN 0-201-18127-4.

[PLRM] PostScript language reference. Adobe Systems Incorporated, 3rd ed. Addison-Wesley, 1999. ISBN 0-201-37922-8.

[INTRO] Display PostScript System. Introduction: Perspective for Software Developers. 15 April 1993.

[CLRM] Display PostScript System. Client Library Reference Manual. 15 April 1993.

[CLSX] Display PostScript System. Client Library Supplement for X. 15 April 1993.

[DPTX] Display PostScript System. Display PostScript Toolkit for X. 15 April 1993.

[PSWRAP] Display PostScript System. pswrap Reference Manual. 15 April 1993.

CONTENTS

1. Introduction 1

2. The DPS client side 1
   2.1 Libraries 1
   2.2 Client-side tools 2
   2.3 Sample clients 2

3. The DPS server side 2
   3.1 Display Ghostscript 2
   3.2 The DPS extension 3

4. References 3

$XFree86: xc/programs/Xserver/hw/xfree86/doc/sgml/dps.sgml,v 1.5 2005/03/04 03:23:52 dawes Exp $