WLPR2.DLL
Version 2.0a

User's Guide

May 27, 1994

Thomas Heil

This document was created with Microsoft Word for Windows 2.0

DISCLAIMER:

THE SOFTWARE IS PROVIDED AS-IS. THE AUTHOR DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. NO LIABILITY IS ASSUMED FOR DAMAGES, DIRECT OR CONSEQUENTIAL, WHICH MAY RESULT FROM THE USE OF THIS SOFTWARE.

(C) Copyright 1992-94 by Thomas Heil.

All trademarks appearing in this document are the property of their respective owners.

Table of Contents

Table of Contents 3
Introduction 5
Copyright: 5
Author: 5
New Features in Version 2.0 6
Setting Properties for Remote Queues 6
DLL Settings 8
    Section Miscellaneous: 8
    Section Timeouts: 8
Known problems: 8
Troubleshooting: 9
Change Log: 9

Introduction

WLPR2.DLL contains a set of API functions that allow an application to print to remote printers, query their queues, and remove print jobs from their queues. These printers must understand the LPR/LPD protocol which is defined in RFC 1179. An application that calls this DLL does not need to have any knowledge about the implementation details. It may call functions in this DLL to set/modify properties for individual queues, such as number of copies to print, whether a banner page shall be generated, etc.

Copyright:

This DLL is Copyright (C) 1993-94 by Thomas Heil. Redistribution of this DLL is allowed only together with the documentation (as PostScript .PS file and as plain ASCII .ASC file). This software may not be sold by itself. Applications that use this DLL may be sold together with the DLL and its documentation included royalty free.

If you want to write a similar DLL that implements the WLPR2 API for another protocol, please feel free to contact me for implementation details.

Author:

Thomas Heil
Hindenburgstr. 50
41352 Korschenbroich
E-Mail: th.heil@kfa-juelich.de (Internet)

Phone: +49 2161 644895
Fax: +49 2161 644337

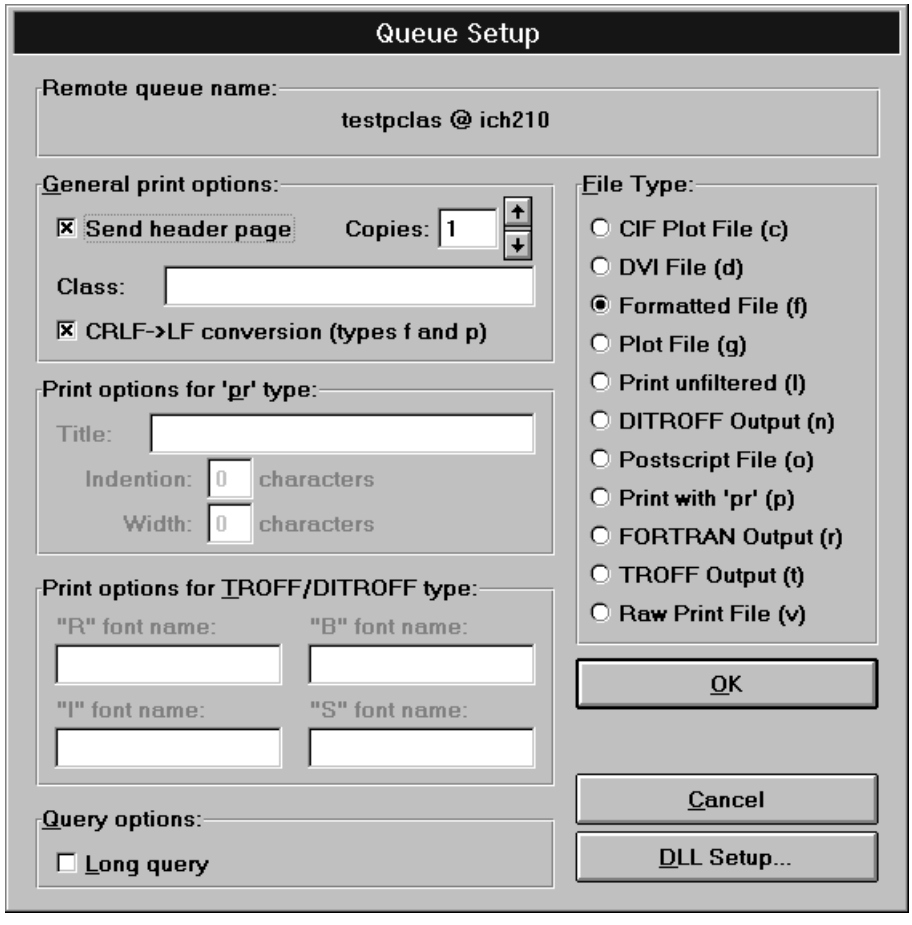
New Features in Version 2.0

The DLL was totally restructured. Now all protocol relevant stuff is totally included in the DLL. E.g. with version 1.x an application had to have an own dialog for setting the queue properties. Now that is all handled by the DLL itself which exports an appropriate function. This makes the calling application independant of the implementation specifics of the transport protocol, and allows it to use several different Protocol DLLs that offer the WLPR2 API side by side.

This DLL is not compatible with WLPR.DLL Version 1.x.

Setting Properties for Remote Queues

If your application allows you to view/change the properties of a remote queue, you get a dialog that looks like this:



In this dialog, the following settings can be made:

Send header page: If checked, a header page is generated with each print job. If not checked, no header page is printed. Note that this might be overridden by the settings of the remote print server (default: on).

Copies: Set the number of copies that a print job generates (default: 1).

Class: Sets the job class of print jobs that go to this queue. This class name might be interpreted by the print server in some way. Check with your local print server administration for classes that have some relevance (default: none).

CRLF->LF conversion Normally, files going to a queue of type "Formatted File (f)" and "Print with 'pr' (p)" run through a filter that translates CRLF sequences to single LineFeeds. This can be switched off with this check box (default: Filtering is on).

File Type: The given file types are those that RFC 1179 defines, although normally just a small subset is really needed. The problem is that not every server accepts all types, so you might have to try a bit around. You should take the following approach: If PostScript files or plain ASCII files are sent to this queue, you should select "Formatted File". If you send binary printer data (e.g. output from a Windows printer driver) to this queue, you should select "Raw Print File". Note that the type "Postscript File" is not accepted by a lot of LPD implementations.

Plain ASCII files can also be sent with the setting "Print with 'pr'". This normally means that the file is sent through some formatting filter on the print server which allows header lines to be written on the output. If you select this type you can specify the desired page header line as "Document Title", and also modify margin and line width settings.

If you select TROFF/DITROFF output you can specify which fonts shall be used for output.

The file types "Formatted File" and "Print with 'pr'" cause the print file to run through a filter that converts CarriageReturn/LineFeed sequences to a single LineFeed.

Long query If checked, the user gets a more verbose listing when querying the remote queue. Otherwise he gets a standard listing.

When the dialog contains just an "Ok" button, the settings are saved when this button is pressed. When the dialog contains an additional "OK / Save" button and the user just presses the "Ok" button, the settings apply only to the action that the calling application is about to perform, but they are not permanently changed. If, in this case, you want to make your changes permanent, press the "Ok / Save" button.

The last button "DLL Setup..." allows the user to influence how the DLL operates in general. These settings are described below in the section ***"DLL Settings"***.

DLL Settings

The following settings that influence the behaviour of the DLL can be made in the "DLL Setup..." dialog. Don't change them unless you think there is a reason to do so (e.g. connection problems).

Section *Miscellaneous:*

Debug mode — If checked, the DLL operates in debug mode. See the section ***"Troubleshooting"*** below for details on this mode. Default is *true*.

Reset connection after LPQ — If checked, the socket connection is reset after an LPQ or LPRM request. This was necessary for some older versions of a few TCP/IP stacks in order to reuse TCP port numbers. Default is *true*.

Linger time — If 0, the "linger" socket option for connections is explicitly set to SO_DONTLINGER.
If >0, the "linger" option is set to SO_LINGER (enabled), and the number specified determines the timeout in seconds.
If -1, the socket's linger setting remains the same as it was when WINSOCK.DLL created the socket.
Default is -1 (linger setting remains unchanged). If you have problems with connections that sometimes work and sometimes not (or fail every other time) set this value to something different than -1.

Section *Timeouts:*

Connect timeout — Time (in seconds) after which a not yet established connection to a server fails. Default is 30.

Send/receive timeout — Time (in seconds) after which a send or receive operation to/from a server fails if there is no progress.

Close timeout — Time (in seconds) allowed for a server to react on a connection closed by the DLL (by closing the connection, too) before the connection is reset.

Known problems:

- Some WINSOCK implementations create problems when a program tries to close a non-blocking socket gracefully. The problem can be circumvented by setting the TCP linger option for the socket connection to a non-zero timeout value in the "DLL Setup..." dialog.
- If queries for a remote queue status hang after a few attempts (or every other try) make sure that the item *"Reset connection after LPQ"* in the "DLL Setup..." dialog is checked.

Troubleshooting:

If you have problems with an application that prints by use of this DLL, please don't hesitate to contact me. But please do one thing before that so that I get some information about what is happening: Switch the DLL into debug mode in the "DLL Setup..." dialog. After this WLPR2.DLL writes a debug log to the file C:\WLPRDEB.LST. (Note that you have to stop all programs using WLPR2.DLL and restart them to get this effective because this INI entry is checked at DLL load time). This file is overwritten with each LPR/LPQ/LPRM request. When you did one of those requests and you got an error, immediately rename the above file to some other name before you execute another request. Then mail me this file. When all problems are resolved you can disable debug mode again.

Change Log:

Version 2.0a:

- WSACleanup() was not called when the WINSOCK.DLL was unloaded.
- Default for the "Reset after LPQ" DLL setting changed to "yes".
- Now the user may explicitly define whether files in queues of type "Formatted file (f)" and "Print with 'pr' (p)" shall run through a CRLF->LF filter.
- If the networking software (or WSOCKAID.DLL, respectively) cannot return a username to WLPR2.DLL (because the networking software does not have something like this and the user did not set one with the WSAIDCNF.CPL Control Panel applet coming with WSOCKAID.DLL), a default username *"pcuser"* is used by WLPR2.DLL. The user is notified of this fact with a message box the first time he tries to print. Later prints use this default without further notification.
- Removed a bug which caused an LPQ request to hang on some WINSOCK implementations.
- All timeout parameters for a connection can now be user-defined in the "DLL Setup..." dialog.