
Harlequin MultiRIP™

Controlling TrapPro™ from PostScript LanguageLevel 3

Technical Note Hqn053

July 2012



This technical note is intended for OEMs supporting the TrapPro™ trapping option to Harlequin MultiRIP™ and Harlequin Host Renderer v3.0r3 and later.

1 Introduction

TrapPro performs automatic in-RIP trapping. TrapPro is supplied as an optional output plugin for the Harlequin MultiRIP. New controls are added to the RIP when TrapPro is installed, such as TrapPro Manager and Ink Set Manager, both available in the Output menu. Refer to the document *TrapPro User Manual* for information on using the TrapPro graphical user interface (GUI).

For the Host Renderer you may embed PostScript LanguageLevel 3 commands that specify trapping options in the PostScript language file, before submitting it to the RIP. This technique requires familiarity with the PostScript language, or an application that can generate the necessary code.

2 Using embedded commands

To understand the background to the technique of using embedded commands in a PostScript language file, see the *PostScript Language Reference* manual, Third Edition, which Global Graphics refers to as [RB3]. In particular, see section 6.3, tables 6.13 and 6.15 (pages 439-454) and the operators **settrapparams** and **settrapzone** (pages 685-686).

Note: This information was first given in a supplement to the PostScript Language Reference Manual for version 3010, [EXTN3010]. If you have that document, which is no longer the definitive reference, see sections 4.5.1 through 4.5.4, tables 4.15 and 4.16 (pages 100-103) and the operators **settrapparams** and **settrapzone** (pages 205-206).

In general, TrapPro supports the trapping zones, trapping parameters, and operators **settrapparams** and **settrapzone** as documented in [RB3]. However, there are some differences, which are documented here. Section 3 on page 4 lists the differences between the LanguageLevel 3 features and TrapPro. Section 4 on page 4 lists Harlequin RIP extensions to LanguageLevel 3.

PostScript language jobs submitted to the RIP can define their own trapping.

The ability for PDF jobs to contain explicit trapping or trapping specifications for creation in the RIP depends upon features in both PDF version 1.3 and the Portable Job Ticket Format (PJTF)

PDF jobs containing PJTF (Portable Job Ticket Format) structures that detail trapping parameters are supported with respect to making the parameters available to TrapPro. Previously, PJTF structures were ignored. If TrapPro is enabled (that is, just the password entered, not necessarily any parameters set via the GUI or PostScript language file), a PDF job containing trapping parameters (via PJTF structures), will be trapped as dictated by those parameters.

PDF jobs can declare whether or not they are already trapped. The Harlequin MultiRIP *does* recognize the `/Trapped` key in the `Info` dictionary of a PDF version 1.3 job and treats a value of `/True` as disabling in-RIP trapping for that job, regardless of any settings in TrapPro.

2.1 Requirements

The Harlequin MultiRIP and Host Renderer installation must have the TrapPro option enabled, for the MultiRIP in the Configure RIP Extras dialog box and the Host Renderer must have a password file.

The use of LanguageLevel 3 (set with **Postscript Language compatibility level** in the Page Setup Options dialog box), is recommended but `settrapparams` is also recognized at LanguageLevel 2.

Some applications can generate appropriate trapping zones and specify trapping parameters, either through built-in features or by using optional plugins, available from the applications' makers or third parties. These applications include:

- Adobe InDesign™
- Adobe PageMaker® version 6.5 and the Adobe trapping plugin
- QuarkXPress®
- Adobe Acrobat®

It is also possible to create custom workflows in which trapping zones and trapping parameters are added to the job before presenting the job to the RIP. For example, an imposition system might create a separate trapping zone around each page in an imposed sheet or flat.

Note: [RB3] says that the result of defining a trapping zone when there are marks on the page is unpredictable and dependent on the implementation. The RIP reports an error if this is attempted, so you must define all zones before marking the page.

2.2 Interaction between settings

There are three places where it is easy to change the trapping options:

- The user interface (GUI) (MultiRIP only)
- Page features
- The job

The only rule of precedence is that the setting made last in time applies. Thus a value set in the job or in a page feature will always override the corresponding setting in the GUI. A page feature applies its settings at the start of each page, so settings in the job have effect only if made for each page. Note, though, that to change trapping parameters for the whole page, that is, to override GUI parameters, the code should force a **setpagedevice** so that the **/Install** hook gets run again. A simple way to do this is to execute:

```
<< /Trapping true >>  
setpagedevice
```

after setting the parameters with **settrapparams**.

An example page feature distributed with the RIP called “Disable Trapping” can be used to disable all trapping settings configured for the RIP and the job.

3 Differences between LanguageLevel 3 and TrapPro

See [RB3] table 6.13 for the LanguageLevel 3 definitions of these parameters. Table 1 describes the differences.

LanguageLevel 3 parameter	TrapPro
ImageInternalTrapping	Ignored.
TrapSetName	Ignored.

Table 1 Differences between LanguageLevel 3 and TrapPro

3.1 Differences between Technical Note #5622 and TrapPro

Previously the default value for the PostScript language parameter **ImageTo-ImageTrapping** (PostScript Language Level 3 Extensions for Trapping—Technical Note #5622), was **false**. In current versions the default has been changed to **true**. This change means that traps, along with a centerline, are generated along an image boundary where the adjacent object is another image. This change overrides the table on page 13 of Technical Note #5622.

4 TrapPro extensions to LanguageLevel 3

This section explains the various PostScript language parameters which can be used with TrapPro:

Note: TrapPro supports the setting and querying of trapping parameter values defined in the [RB3]. However, you should note that some of those parameters are not used with TrapPro so the values given to them are irrelevant.

4.1 TrapPro configuration file

Shown below is an example of PostScript language configuration file for TrapPro:

```
% OEMs will need to set this password:
% << /TrapPro 0 >> setsystemparams

<<
  /BlackWidth 1.0
  /Enabled true
  /HqnDefaultZone false
  /HqnObjectToImageRule /Normal
  /HqnSmallObjectProtectionFactor 0.6
  % /HqnHighlightTrapColor [0.0 1.0 1.0 0]
  % /HqnTextToImageRule (Spread)
  % or
  % /HqnTextToImageRule (Normal)
  /HqnTrapAspectRatio 1.0
  /HqnTrapFeathering 1
  /ImageInternalTrapping false
  /ImageToObjectTrapping true
  % /ImageTrapPlacement /Center
  % or
  % /ImageTrapPlacement /Normal
  /SlidingTrapLimit 0.9
  /StepLimit 0.1
  /TrapColorScaling 1.0
  /TrapWidth 0.516
>>
/Trapping /ProcSet findresource /settrapparams get exec

<<
  /Install [
    currentpagedevice /Install get /exec load
    {
      newpath /Trapping /ProcSet findresource /settrapzone get
exec
    } bind
    /exec load
  ] cvx
>>
setpagedevice
```

The following table lists the extensions to support TrapPro in the Harlequin MultiRIP and Host Renderer v3.0r3 and later.

TrapPro parameter	Type	Semantics
HqnDefaultZone	Boolean	<p>(<i>Optional</i>) If true, indicates that any zones defined will be deemed as “default” in the terms defined by the RB3 in Section 6.3, In-RIP Trapping. Typically, they will be those defined during the /Install hook of set-pagedevice, and to that end, the RIP internally sets this trapparam to true while it is running /Install (and sets it back to false afterwards). See also Section 4.2, 'Additional notes'.</p> <p>Note to OEMs: Normally, you will not need to change this value.</p> <p>Default: False</p>
HqnEffort	Integer	<p>(<i>Optional</i>) An integer value which indicates to the trapping engine how much computational effort to invest in trapping. As the value rises, more computationally (memory and/or time) intensive algorithms will be used. Users may notice better trap quality with higher values of this parameter in more complex jobs.</p> <p>Default: 0</p>

Table 2 TrapPro extensions to LanguageLevel 3

TrapPro parameter	Type	Semantics
HqnHighlightTrapColor	Array of four numbers	<p><i>(Optional)</i> If present, traps are drawn in the specified CMYK color, <i>in over-print mode after the normal traps</i>. Also, a dashed line is drawn around the outline of the zone, using the same separation and color.</p> <p>Highlight traps for each trapping zone are drawn in their own separation (pagebuffer) if the current color separation Style (CSS) is set to autoseparate. This makes it possible to use the other pagebuffers to produce normal output without retrapping.</p> <p>Range of each number: 0 - 1</p> <p>Default: no highlighting</p>
HqnObjectToImageRule	Name or string	<p><i>(Ignored)</i> You can set and query this value but it is ignored by the trapping engine.</p> <p>Default: Normal</p>

Table 2 TrapPro extensions to LanguageLevel 3

TrapPro parameter	Type	Semantics
HqnPoorTrapRule	Boolean	<p>(<i>Optional</i>) When true, compares “colorant concentrations” as defined in the “Normal Trapping Rule” description in section 6.3 of the RB3.</p> <p>When false, compares neutral density contributions per colorant. This has the effect of making more traps where the difference in neutral density at a boundary is large, and fewer when it is small. This better achieves the fundamental aim of trapping, namely to prevent “light leaks”, whilst resisting the temptation to insert traps where they are not necessary to achieve this aim (especially at boundaries differing only in the yellow colorant).</p> <p>Default: false</p>
HqnSmallObjectProtectionFactor	Number	<p>(<i>Optional</i>) This option allows you to protect small objects on the page by reducing the trap width when trapping against them. The protection factor that you enter ensures the trap width is never greater than a ratio of the average width of the object to the normal trap width. For example, a value of 0.6 ensures the trap is never greater than 60% of the object it is trapping against.</p> <p>Default: 0.6</p>
HqnTextToImageRule	Name or string	<p>(<i>Ignored</i>) You can set and query this value but it is ignored by the trapping engine.</p> <p>Default: Normal</p>

Table 2 TrapPro extensions to LanguageLevel 3

TrapPro parameter	Type	Semantics
HqnThinnessLimit	Number	<p><i>(Optional)</i> Defines a limit for object thinness beyond which an object will not be trapped. It is not necessarily useful to attempt to trap hairlines and other thin objects such as sparsely populated imagemasks and very small characters. The smaller this value is set, the more traps will be suppressed. The default value is such that only hairlines (1 device pixel wide) and equivalent shapes will be left untrapped.</p> <p>This does not work on autochoked objects.</p> <p>Default: 2.0</p>
HqnTrapAspectRatio	Number	<p><i>(Optional)</i> The ratio of all brush heights to widths. This is the parameter which controls anamorphic trapping. The default value is 1.0, which means the amount of trapping applied is equal in the x and y axes (a circular brush), but by varying this parameter you can use an elliptical brush to apply differing amounts of trapping in the two directions.</p> <p>Default: 1.0</p>

Table 2 TrapPro extensions to LanguageLevel 3

TrapPro parameter	Type	Semantics
HqnTrapFeathering	Integer	<p>(<i>Optional</i>) Indicates the number of feathering steps to be applied during trap generation. Feathering turns the hard edge of a normal trap into a gradation which softens the area of the trap, reducing its visual impact. This provides a gradually reducing amount of compensation for possible mis-registration rather than the all-or-nothing approach of a solid-colored trap. The default value is 1, meaning a normal, constant-color trap. Note that feathering is achieved at the expense of considerable extra computation, so increase this value with caution.</p> <p>Default: 1</p>
HqnTrapOverprintedObjects	Name or string	<p>(<i>Optional</i>) Controls which, if any, overprinting objects are considered for trapping.</p> <p>Values:</p> <p>/All: trap everything regardless of overprint setting.</p> <p>/NotBlack: overprinted 100% black (actually, anything matching the black rule) does not trap, everything else does.</p> <p>/None: no traps for overprinting objects.</p> <p>Default: /All</p>

Table 2 TrapPro extensions to LanguageLevel 3

TrapPro parameter	Type	Semantics
HqnTrappingCellSize	Integer	<p>(<i>Optional</i>) A size, in KiB, that the trapping code attempts to use as its fundamental cell size. The larger the value, the bigger the cell leading to fewer cells required to trap a page. Also, the overhead arising from the infrastructure and the necessary overlap between adjacent cells is minimized.</p> <p>A smaller value means more cells and more overhead, but it also increases the chances that a cell can be completely populated by values from the backdrop store during compositing, which means the cell can then be trapped and freed. That is, a smaller cell size might mean a lower memory high water mark during compositing.</p> <p>Therefore, If a trapping job is performing heavy compositing and running out of memory causing a WMError, especially at medium or high resolutions, try reducing the cell size to a value much less than the default. A good starting value is 128.</p> <p>Default: 1024</p>

Table 2 TrapPro extensions to LanguageLevel 3

TrapPro parameter	Type	Semantics
HqnTrappingMemoryLimit Notes: This parameter is superseded by "HqnTrappingCellSize" on page 11.	Number	<p>When the usage of memory by TrapPro exceeds this limit, large structures will be written to disk and then freed when TrapPro is using less than the limit. It is therefore a "soft" limit, in that memory usage can go above it in normal operation: it is a threshold value, above which, trapping considerations are written to disk instead of allocated trapping memory.</p> <p>The maximum memory is 2GB.</p> <p>When the trapping memory limit has been exceeded and the structures are starting to be written to disk, the following message appears in the monitor and the log file:</p> <pre>%%[Warning: TrapPro memory limit (256mb) exceeded]%%</pre> <p>The figure in parentheses reflects the current setting. When this warning appears performance may be slow. Therefore, if more RAM is available, you should consider raising the value.</p> <p>HqnTrappingMemoryLimit should be set within a file in ExtraStart. As it is a sysparam, it will persist for all page setups.</p> <p>Default: 256</p>

Table 2 TrapPro extensions to LanguageLevel 3

TrapPro parameter	Type	Semantics
HqnTrapSizeSenseFactor	Number	<p>(<i>Optional</i>) This is a floating point number which is a multiplier. It is the ratio of mask brush size to normal brush size, which essentially means that the larger the number, the further the engine looks for areas of color away from which a potential trap should narrow. The default value of 2.0 results in a mask brush twice as large as the final trap brush, and therefore any trap will occupy at most one half of the area available to it.</p> <p>Default: 2.0</p>
HqnWhiteDensityLimit	Number	<p>(<i>Optional</i>) When a color's calculated total neutral density is less than this figure, it is treated as white. This means that a superblack will auto-choke against a background of this color, and the color itself cannot spread. The parameter is provided to allow for some adjustment, particularly in situations where a zero tint in all separations is not the only sensible definition for white, for example, when "page white painting" is enabled in a color managed setup.</p> <p>Default: 0.005</p>
Override	Integer	<p>(<i>Optional</i>) When override is set to an integer, subsequent calls to settrapparams are ignored unless their dictionary operands also contain an Override key with a value greater than that already set.</p> <p>Default: not present</p>

Table 2 TrapPro extensions to LanguageLevel 3

4.2 Additional notes

If a default trap zone (**HqnDefaultZone**) is defined with an empty path (as now happens in the RIP when TrapPro is enabled via the GUI), it is overloaded by the RIP and the device bounds at the end of the **BeginPage** hook are substituted.

Change history		
v3.0	2012.06.19	Updated for MultiRIP v3.0r3.
v2.7	2010.04.14	Add note to HqnTrappingMemoryLimit (superseded).
v2.6	2010.03.04	Add HqnTrappingCellSize .
v2.5	2010.02.19	Added information about the ImageToImageTrapping parameter.
v2.4	2007.05.17	Added more to HqnTrappingMemoryLimit .
v2.3	2006.08.02	Updated HqnTrappingMemoryLimit entry and copyright text.
v2.2	2005.01.17	Updated copyright and logos.
v2.1	2004.11.11	Added Hqneffort and HqnTrappingMemoryLimit .
v2.0	2003.10.14	Harlequin RIP Eclipse Release SP3 added HqnTrapOverprintedObjects and HqnThinnessLimit . Update with PJTF information. Updated copyright.
v2.0	2003.04.08	Change to HqnThinnessLimit .
v 1.0	2002.11.26	Harlequin MultiRIP Eclipse Release.



Copyright and Trademarks

July 2012

Document issue: 307

Copyright © 1992–2012 Global Graphics Software Ltd.

Certificate of Computer Registration of Computer Software. Registration No. 2006SR05517

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 Global Graphics Software Ltd.

The information in this publication is provided for information only and is subject to change without notice. Global Graphics Software Ltd and its affiliates assume no responsibility or liability for any loss or damage that may arise from the use of any information in this publication. The software described in this book is furnished under license and may only be used or copied in accordance with the terms of that license.

Harlequin is a registered trademark of Global Graphics Software Ltd.

The Global Graphics Software logo, the Harlequin at Heart Logo, Cortex, Harlequin RIP, Harlequin ColorPro, EasyTrap, FireWorks, FlatOut, Harlequin Color Management System (HCMS), Harlequin Color Production Solutions (HCPS), Harlequin Color Proofing (HCP), Harlequin Error Diffusion Screening Plugin 1-bit (HEDS1), Harlequin Error Diffusion Screening Plugin 2-bit (HEDS2), Harlequin Full Color System (HFCS), Harlequin ICC Profile Processor (HIPPP), Harlequin Standard Color System (HSCS), Harlequin Chain Screening (HCS), Harlequin Display List Technology (HDLT), Harlequin Dispersed Screening (HDS), Harlequin Micro Screening (HMS), Harlequin Precision Screening (HPS), HQcrypt, Harlequin Screening Library (HSL), ProofReady, Scalable Open Architecture (SOAR), SetGold, SetGoldPro, TrapMaster, TrapWorks, TrapPro, TrapProLite, Harlequin RIP Eclipse Release, Harlequin RIP Genesis Release, Harlequin MultiRIP, Harlequin Parallel Pages and Harlequin VariData are all trademarks of Global Graphics Software Ltd.

Protected by U.S. Patents 5,579,457; 5,808,622; 5,784,049; 5,862,253; 6,343,145; 6,330,072; 6,483,524; 6,380,951; 6,755,498; 6,624,908; 6,809,839.

Other U.S. Patents Pending

Protected by European Patents 0 803 160; 0 772 934; 0 896 771; 672 29 760.8-08.

Portions licensed under U.S. Patent No. 5,212,546; 4,941,038.

TrueType is a registered trademark of Apple Computer, Inc.

The ECI and FOGRA ICC color profiles supplied with this Harlequin RIP are distributed with the kind permission of the ECI (European Color Initiative) and FOGRA respectively, and of Heidelberger Druckmaschinen AG (HEIDELBERG).

The IFRA ICC profiles supplied with this Global Graphics Software are distributed with the kind permission of IFRA and of GretagMacbeth.

International Cooperation for Integration of Processes in Prepress, Press and Postpress, CIP4, Job Definition Format, JDF and the CIP4 logo are trademarks of CIP4.

Adobe, Adobe Photoshop, Adobe Type Manager, Acrobat, Display PostScript, Adobe Illustrator, PostScript, Distiller and PostScript 3 are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries which may be registered in certain jurisdictions.

Global Graphics Software Ltd is a licensee of Pantone, Inc. PANTONE® Colors generated by ScriptWorks are four-color process simulations and may not match PANTONE-identified solid color standards. Consult current PANTONE Color Publications for accurate color. PANTONE®, Hexachrome®, and PANTONE CALIBRATED™ are trademarks of Pantone, Inc. © Pantone, Inc., 1991.

Other brand or product names are the registered trademarks or trademarks of their respective holders.



US Government Use

Harlequin MultiRIP software is a computer software program developed at private expense and is subject to the following Restricted Rights Legend: "Use, duplication, or disclosure by the United States Government is subject to restrictions as set forth in (i) FAR 52.227-14 Alt III or (ii) FAR 52.227-19, as applicable. Use by agencies of the Department of Defense (DOD) is subject to Global Graphics Software's customary commercial license as contained in the accompanying license agreement, in accordance with DFAR 227.7202-1(a). For purposes of the FAR, the Software shall be deemed to be 'unpublished' and licensed with disclosure prohibitions, rights reserved under the copyright laws of the United States." Global Graphics Software Incorporated, Somerset Court, Suite 320, 281 Winter Street, Waltham, MA 02451.