

Defining And Setting Up Those Darn Printers And Drivers!

Kevin Gillins

GLC Consulting Group, Inc.

kevin.gillins@glcgroup.com

Introduction

This paper will share with you my experience with printing and Oracle Applications over the last seven years. I started my experience with Oracle Applications release 9.3.7 using the Oracle RDBMS version 6. Since this time I have moved through the various releases up to my most recent experience of Applications release 11.5.3 (11i) using Oracle RDBMS 8.1.6.

Included at the end of this paper are various screen shots from the Oracle Applications demonstration software version 11.0.3. Unfortunately I was not able to get screen shots of 11i for this paper.

Abstract

Each area of printer setup utilizing Oracle Applications will be discussed. This includes printer styles, printer types, printer drivers, and the association of each component. Registering printers and the parameters of the concurrent program will also be reviewed.

Overview

This paper will provide you with an understanding of printers, printer types, printer styles, and printer drivers. You will find out the mechanics of how Oracle Applications processes the print requests and how you can control the output for your printers. With the information from this paper, you will be able to adapt your printers to the Oracle Applications. Also, you will learn how registering concurrent programs in the system will impact the printer driver selection for the program.

Additional information will be provided on how printer drivers can be used to process the output through external programs. This technique can be useful for FTP and other file processing tools. For example, you may have a report archive utility that you need the reports to go to. By defining a printer driver you would be able to invoke the report archive tools or, you may want the report to be published on a web site and placed in the web directory for access. The utilization of a custom registered profile with a custom printer driver you can secure printers to be used by specific users. This may be useful in conjunction with the HTML idea mentioned. Furthermore, you may adapt these ideas to automate email distribution of reports. At a minimum you will come away with an understanding of how the printer process works within the Oracle Applications.

Components of the printing process

There are several pieces necessary to have in place before you will be able to print reports from Oracle Applications. The pieces that are needed in order to print are: UNIX print queues, printer styles, printer drivers, printer control files, and printer types, all of which will be discussed in more detail. While it is possible to generate output from the software right out of the box you will find that fine tuning and efficient use of printers and paper will require changes to the printer configuration delivered.

Styles

The Printer Style is responsible for controlling the dimensions of the report. When the Concurrent Program is one of Oracle Reports and the number of copies is zero, the Printer Style determines which printer control file is used. The Printer Style also determines the dimensions of the report. The dimensions set up on a style are matched against the dimensions defined for the program. The styles that allow the report to be produced without going beyond the dimensions defined will be made available to the user. Several examples of printer control files can be found in Figures 1, 2, and 3. Depending on your printing environment you may need to define the “return” entry in the printer control file with a space rather than “control (J)”. In cases where your report prints the first line and the remainder of the report is blank, you may want to experiment with this setting.

Field by field. (Fig 6)

Style Name: Style name is the internal name used by the applications to identify the Printer Style in the system.

Seq: Sequence number is not relative in the process of printing. It is used here to uniquely identify the print styles.

User Style Name: User style name is the user-friendly name defined by the user. It should be a meaningful name that can be recognized by the end users of the system. A name should be selected that best represents the style and orientation.

SRW Driver: SRW (SQL Report Writer) driver is where the reference to the printer control file is made. This reference is the exact name of the printer control file without the extension of “prt”. You can find the printer control files located in \$FND_TOP/reports for release 11 and \$FND_TOP/srw for release 10 installs. Fig 1 is an example of the HPP.prt control file for portrait orientation.

Description: Description documents and gives a meaningful long description to the Printer Style.

Columns: The value entered should be the maximum allowed number of columns for the printer driver that will be associated for printing. There are standard dimensions for printers that can be followed. I generally never need to make any changes to this definition. This value should be the same as the width value in the printer control file depicted in the SRW field.

Rows: The value entered should be the maximum allowed number of rows for the printer driver that will be associated for printing. There are standard dimensions for printers that can be followed. I generally will make changes to this value to match the lines per inch and correct page overflow on reports. This value should be the same as the height value in the printer control file depicted in the SRW field.

Suppress Header: Suppress header is used to send a control code to the printer queue, and not to print the banner page of a report.

Orientation: Orientation field has no impact on the printing process from what I can determine. The actual orientation is set in the printer driver definition at the initialization field.

Drivers

The printer driver is used when the report is actually being printed to the queue. The printer driver defines the initialization process and other formatting items, such as fonts and margins. A printer driver is invoked when the printing of the report actually takes place. The printer driver defines the utility or program that is responsible for processing the output to the printer queue. The printing utility or program will only be invoked when the number of copies specified is greater than zero at the time the report or reprint is requested.

The printer driver provides you with the option of specifying the utility or program, such as a UNIX shell script or other created program. The program specified can be a custom created program. With this program, you will have access to the report output for post processing. You could have a fax server, report archive, or even email processing system where the program identifies who to email the report to.

Field by field. (Fig 7)

Driver Name: Driver name is the system driver name you want to assign to the driver definition. I suggest that any and all drivers that you want to modify you actually copy and create a new name for it. Use a prefix to the name that identifies your customization of the printer drivers. This will help preserve your customizations during future upgrades and serve as a quick way to identify the printer drivers created on site from the ones delivered by Oracle.

User Driver: User driver is a user-friendly name given to the printer driver. This name should also use some kind of prefix to help identify your custom drivers.

Description: Description has no processing logic. Use this field to describe your custom printer driver. Special comments should be added to help you remember what you changed or created in the driver.

SRW Driver: SRW Driver is the name Oracle Reports driver that will be used to format reports. This SRW driver will only be used when the number of copies requested is greater than zero. Also note that this driver is used at report generation time and will not be used during reprints. If you generate a report using a HP type printer driver and then issue a reprint to another printer that uses a LINE printer driver your report will not be printed using the LINE printer driver.

Platform: Platform is a list of predefined typical platforms used. It is suggested that you leave this field NULL. When the printer driver is executed the Concurrent Manager will select the driver with the platform name. If one is not found, then the NULL platform becomes the default.

Driver Method: Driver method can be specified as Command, Program, or Subroutine. When you select Command, then the command found in the Arguments field will be used. When you select Program then a program you specify or create will be executed instead of the Arguments field. The program must be found in the PATH variable and be available to the Concurrent Managers. When you select Subroutine the printing is handled by an operating system call, which is platform specific.

Driver Method Parameters: Driver method parameters can have options of Spool File and/or Standard Input and a place for Program Name. If you want to have a separate spool file created then check this option when the report is required to be a file rather than standard input. You would select Standard Input if you have selected program for the method and the program accepts standard input. If your program does not accept standard input data then deselect this option. The program name is where enter your program or subroutine name. Command line parameters can be specified for programs in this same field.

Arguments: Arguments is where the operating system shell command is specified along with the parameters required. Profile values can also be accessed here. If the shell command being executed has standard output for error messages, you can find the standard output in the concurrent manager log file. In order for the shell command to issue error messages to the actual request log, the shell will need to determine the name of the file first. This technique is useful for custom shell commands that issue error or warning messages and where you want those messages to be seen by the user.

Initialization: Initialization field is where the printer commands are stored for your specific printer. The commands placed in this field are sent to the printer before any report output is sent. Also, it is important to note that these initialization values are sent to the printer only when the number of copies is greater than zero. Typically the values that go in this field are escape codes for HP style printers using the PCL command language. Much of the effort during the printer setup will go into getting this set of PCL commands correct.

Reset: Reset has a similar purpose as the initialization. The difference is the values in this field are sent to the printer following the report output. The printer command code in this field is generally a reset to clean up any settings the printer has in effect. Sending a reset command prepares the printer for the next print job that will be processed.

Types

The Printer Type is a logical definition. This can match the printer models that you have in your company. There may be a set of drivers that can be used with several printers in your organization. By defining a printer type for all of these printers you will simplify the setup and maintenance of printer drivers. One printer can have only one printer type assigned to it. When a report request is made the user will select a printer. The printer is associated with a printer type. Defined under the printer type are printer styles and drivers. Only the printer styles defined for the given printer type will be available to the user. Further, only the printer styles that match the report dimensions being requested will be made available. The printer type does not impact the actual printing of reports but rather it controls the styles and drivers that are available when making a report request. Starting in release 10.7 the printer driver can not be modified while it is associated with a printer style. You first had to remove the association before the printer driver could be modified. Once your modifications are complete you need to re-associate the driver with the style to make it available for use.

Field by field. (Fig 5)

Type: Type is the name of the printer type that you want to define. The system comes with many types already defined. This name should be short and clearly identify the printer groups desired.

Description: Description is a field used to describe the printer type you have. You can use this field to clearly document the printer type you are creating.

Style: The style field is the same style that was discussed previously. Select the style you want to make available in this printer type. The style defines the dimensions of the report.

Driver Name: The driver name is the same printer driver that was discussed previously. Select the printer driver desired for the associated printer style. You can have as many printer style and driver combinations as necessary within the printer type; however, the printer styles must be unique within the printer type. You can reuse the printer driver with different styles if necessary.

Printer Control Files (PRT)

Oracle Reports and Concurrent Managers use the printer control files to understand how to handle special control characters on reports. The printer control files (PRT) replaced the printer definition files (PDF) that were used in release 9 of the Oracle Applications. You can locate the printer control files under the \$FND_TOP/reports directory in release 11 and \$FND_TOP/srw directory for release 10. If you are release 9 then the files are under the \$FND_TOP/bin directory and have an extension of "pdf". Printer control files contain the page control, the special printing commands, and the escape codes. The printer control file is used in the printer style and printer driver definition. It is important to note that when the number of copies is equal to zero, then the printer control file from the printer style will be used. When the number of copies is greater than zero, then the printer control file from the printer driver is used. Also note that only during the original report generation will the printer control file be used. If you run a report with zero copies and then perform a reprint, the printer control file definition from the original request will be used for report formatting. If you request a report that formats based on a printer control file from the printer driver then be sure to specify greater than zero copies. Some examples of printer control files can be found in Figures 1, 2, and 3.

Another important point about the printer control files is that the height and width values should match the values defined on the printer style for rows and columns. The consistency of these values will assure consistent results from report output.

Field by field. (Fig 1)

Printer: This is the name of a generic type of printer in double quotes.

Height: Height is the same as rows in the printer style. These values should match the dimensions of reports. In SQLPLUS the PAGESIZE option defines the number of rows.

Width: Width is the same as columns in the printer style. This value should be typical sizes for reports. Generally there are three sizes used: 80, 132, and 180.

Between pages: Between pages indicates the control code to generate between pages on the report. This is the new-page code for the printer.

Return: Return is the control code used to issue a carriage return on the printer. The Oracle Reports will use this value in combination with the linefeed value to generate a next line on the report.

Linefeed: Linefeed is the control code used to issue a linefeed on the printer. The Oracle Reports will use this value in combination with the return value to generate a next line on the report.

Code: Code can be repeated as many times as necessary. The name of the code will appear in double quotes and then the control code necessary to perform the desired action will follow. In the printer control files, there are generally four codes defined. Each of these codes can be used in Oracle Reports to further enhance and control the report output.

Register printers

The *Register Printer* form is where an actual UNIX printer queue name is linked to the Oracle Applications. The printer is assigned to a printer type and then given a useful, user meaningful/friendly description. The printer name in the applications is required to be the same as the UNIX queue name in order to have the output directed to an actual printer. The named registered is case sensitive to the UNIX queue name and must match exactly in case and characters. Also note that the printer registered can not be deleted. For example, define a printer type named obsolete. In this printer type do not define any printer styles and drivers. When you have a printer queue that becomes obsolete you can change the printer to this printer type. This will prevent people from using a printer that is registered but obsolete. When the user selects the printer they will not have any style options available, which will effectively disable the printer.

It is possible to use a fictitious name that may be meaningful to the end user. It would then be necessary for the printer driver to send the output to a realistic print queue name.

Field by field. (Fig 4)

Printer: This is the name of the printer. It must match the print queue name. This value is case sensitive on the UNIX platform and must match character by character.

Type: Type is the printer type as discussed previously.

Description: Description should assist users to identify the printer and location.

Concurrent programs

Concurrent program definitions have an area called Output. This information is where all of the attributes necessary for the program output formatting are defined. This is the only area that will be discussed here. The minimum and maximum number of columns, minimum and maximum number of rows are defined in the Output area. This is the number of columns and rows in terms of character positions, not pixels or other measurements. At the time a request is made, the values defined for columns and rows are used. The system will compare the columns and rows defined for a concurrent program against the columns and rows in a printer style. Only printer styles that allow the report to fit in the dimensions will be made available to the user. To override or force other printer styles you can define a default style at the report level. If you want to lock the style so the user can not change it, then select the style required option. Further, if you want to only allow a specific printer to be used, like check printing, you define the printer name. At the concurrent program level, there is a format option new in release 11. I have not researched what impact this option has on the report.

If you want to have the concurrent program generate graphic output then you need to use the options field in the program definition. In the Executable area of the concurrent program registration you can add options to the program. This should only be done where the program is expecting the options. For Oracle reports that you create in a bitmapped format you could specify the options of “version=2.0b ORIENTATION=LANDSCAPE PAGESIZE=7.5X9”. Please refer to Oracle documentation for further details concerning bitmapped reports.

Field by field. (Fig 8 Output area only)

Format: Format is a value that helps determine the report-viewing tool used to view the report. If a report generates HTML then you should select a format of HTML. There is a profile option for viewing HTML that will be selected depending on this option.

Save: Save option specifies if the Concurrent Managers should save the output from this program. If this option is not selected, then any output generated will be deleted at the end of the program execution.

Print: Print option specifies to the Concurrent Manager if the report should be made available to print. If this option is deselected then the user will not be able to print the output from this request.

Columns: Columns is the number of characters that the program will generate. This setting should be set to the maximum your program will generate. If you have a program that generates several reports and the widest report is 125 characters, then you should set this option to 125. Oracle will determine the proper print style to use for printing this report.

Rows: Rows is the number of lines that the program will generate per page. This setting should be set to the maximum number of lines you expect your report to create on a page. This is not the number of lines for the entire report but rather, the number of lines per page of the report. Oracle will determine the proper print style to use for printing this report.

Style: Style is the print style you want to define as default. When the user makes a report request the system will use this style as the default on the request. If you leave this field blank, then Oracle will determine a style to use based on the dimensions of the report.

Style Required: Style required option, when selected, will force the user to use the printer style defined. The user will not be able to change the printer style at request time.

Printer: Printer defines the printer that this request is to be printed on. When this field is populated, then Oracle will use this printer for producing the output.

Profile values

With release 11, the number of profile values that impact reporting have increased. I will list just a few of the key ones here. The screen prints given are more inclusive of the profile options available. There could be more than reviewed in this paper that can impact printing. I have yet to uncover all of the options. Please refer to the Oracle documentation for detailed explanation of the profile options and their use.

There is a list of profile options that start with the “Viewer%” name (Fig 9). These profile options are used during the report viewing process. They do not impact the actual report generation.

Another list of profile options start with the “RRA%” name (Fig 10). These profile options are used during the report review process and do not impact the actual report generation.

There is a large list of profile options with the name starting like “Concurrent%” (Fig 11 and 12). Some of these profile options will impact the report output. Profile options are defaults and can be overridden at request time. One such profile is “Concurrent:Report Copies”. This profile option can be set to any positive integer. I recommend that you set it to zero at the site level and then allow individual users to change the number of copies as required.

One last profile option is “Printer”. This again is a default printer and should be set to a NULL printer at the site level. Each user can then change their personal profile option to a printer they desire to use for output.

A report viewing software that works well for users that like to switch the report-viewing tool frequently. A product called FTView that allows you to configure up to five buttons to view a report. Each button can open a viewer of your choice with the report output. Some of the users I have shown this tool to have enjoyed it. One client has used this tool for performing desktop post processing.

Sharing ideas

I now have an example of a use for custom printer drivers. I had a client that required report archiving to a report archive tool. The tool was called StoreQM. The requirement was that each report had to have a name associated with it. As the output was passed to the report archive tool it needed to know that the report name was. This required some work to figure out for Oracle Applications. Further, the client indicated that only authorized users should be allowed to generate reports to the archive tool. After researching the clients needs, I determined that we would create a special printer driver for each report that was defined in the report archive. The client liked this idea because it gave them visibility to the reports defined as archive reports. While using the application, the user does not see the driver name. As a result, we defined a printer style to go with it. Each printer style was named the same name as the report archive tool used. Now, when users requested a report and wanted it to go to the report archive tool they select the appropriate printer style.

What was done to implement this process? First, a UNIX shell script was developed that would accept the appropriate parameters from the Concurrent Manager as a print request (Fig 14). You can see that there are several parameters passed to the shell script. These parameters come from the argument field of the printer driver (Fig 13). Each printer driver setup would pass the parameters to the shell script being executed. Also you may see that there is a custom profile name being passed. This profile name is used for the requirement to limit which users can send reports to the archive. The system administrator will maintain the custom profile for each user by setting the profile to Yes for specific users and leaving it at a default of No for everyone else. When an unauthorized user attempts to archive a report they receive an error message in the report request log. This solution has worked well for my client and gave me the foundation for other ideas using this technique.

Settings that work for me

Over the years of experience with trial and error, I usually will start with the following changes to the printer style, printer driver initialization strings and the printer control files. If you are running a typical HP5, HP4, or HP3 then you should start with the following settings to get started. Remember to shutdown and restart the Concurrent Managers after making changes to the printer configuration information. This information is stored in cache memory and is not reloaded until the Concurrent Managers are started.

Make the following changes to the printer control files. The following is a matrix showing the parameters in the control file to change.

Control file	Settings	
HPP.prt	width	80
	height	65
	return	“ “
HPL.prt	width	132
	height	57
	return	“ “
HPW.prt	width	180
	height	65
	return	“ “

Next you need to modify the printer styles for dimensions to match the printer control files. Also change the SRW driver parameter to use the modified printer control files above. The following matrix gives you the information for each printer style.

Style Name	SRW Driver	Columns	Rows
Portrait	HPP	80	65
Landscape	HPL	132	57
Landwide	HPW	180	65

Now define new printer drivers with fresh initialization strings. The following matrix will document the printer driver fields. For a detailed explanation of the PCL codes refer to the HP web site.

I generally will copy the HPLJ4 printer drivers and add a custom prefix to the name so I can identify the custom drivers from those delivered by Oracle.

Driver Name	SRW Driver	Driver Method	Driver Method Parameters	Initialization
CUST_PORTRAITHPLJ4	HPP	Command	Standard Input	/eE/e&10o2a7C/e(s0t0p11H/e&a5L/e&k2G
CUST_LANDSCAPEHPLJ4	HPL	Command	Standard Input	/eE/e&11o2a5.25C/e(s0p12h8.5V/e&k9.5H/e&k2G
CUST_LANDWIDEHPLJ4	HPW	Command	Standard Input	/eE/e&11o2a5.25C/e(s0t0p17h8.5V/e&k7H/e&k2G

To understand the PCL language codes used in the PCL screen you can go to HP web site for details. Perform a search for “PCL commands” and you should find resources and references to help.

<http://www.hp.com>

The final step is to build the printer type and associate the new printer driver with the modified printer style. Build a new printer type that has a custom name to refer to the new custom printer drivers. You then will need to assign the printer queues to the printer type setup.

About the Author

Kevin Gillins has worked in the field of Information Systems for over 16 years. He has developed systems as well as supported Dun and Bradstreet financial systems. The last eight years Kevin has implemented supported and developed Oracle Applications. Kevin's implementation experience is vast. His client experience ranges from single module implementations to virtually all modules implemented. He has also done extensive work with legacy mainframe systems, as well as development complex conversion routines for GL and interfaces with other legacy systems. His assignments have required learning and understand of the printing process extensively. Most clients are accustomed to sophisticated printing capabilities and getting Oracle Applications to follow suit can be challenging.

```
printer "HPLJ Portrait"
```

```
height 66  
width 80
```

```
between pages control(L)  
return control(M)  
linefeed control(J)
```

```
code "bold on" esc "(s7B"  
code "bold off" esc "(s0B"  
code "underline on" esc "&d0D"  
code "underline off" esc "&d@"
```

Fig 1 Sample PRT file (HPP.prt)

```
printer "HPLJ Dynamic Portrait"
```

```
height 66  
width 80
```

```
return control(M)  
linefeed control(J)
```

```
code "bold on" esc "(s7B"  
code "bold off" esc "(s0B"  
code "underline on" esc "&d0D"  
code "underline off" esc "&d@"
```

Fig 2 Sample PRT file (HPPD.prt)

```

printer "HPLJ Landscape"

height 57
width 132

between pages control(L)
return control(M)
linefeed control(J)

code "bold on" esc "(s7B"
code "bold off" esc "(s0B"
code "underline on" esc "&d0D"
code "underline off" esc "&d@"

```

Fig 3 Sample PRT file (HPL.prt)

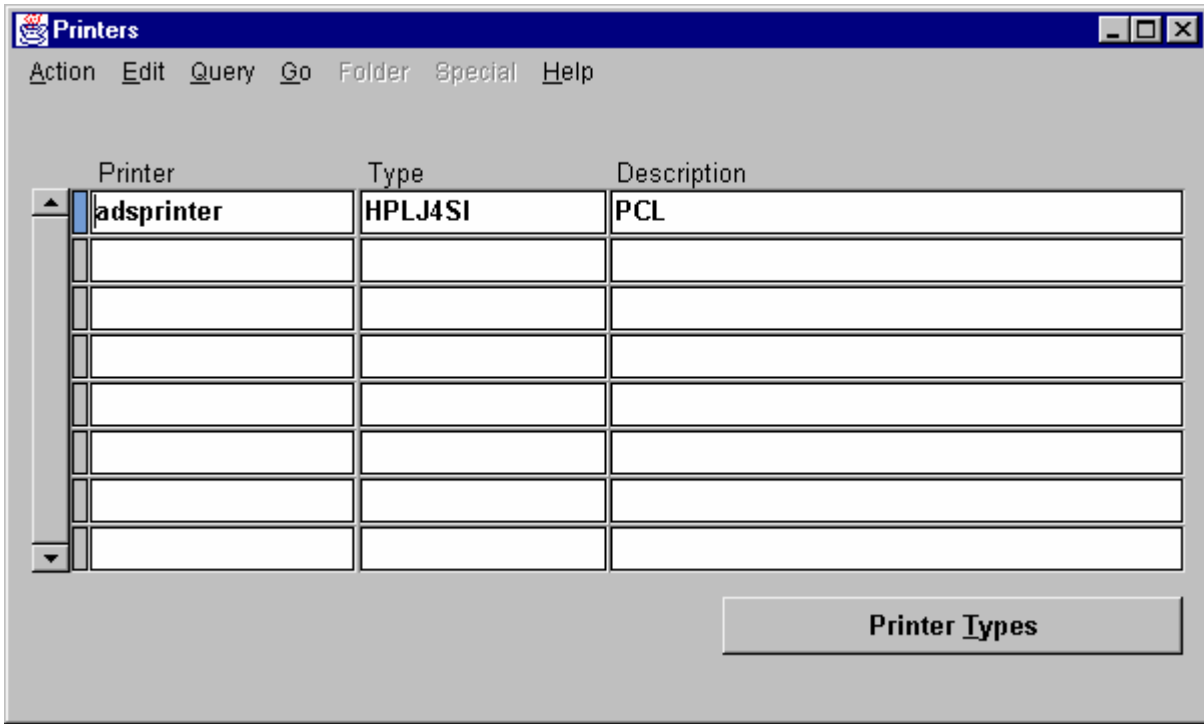


Fig 4 Sample of the register printer window.

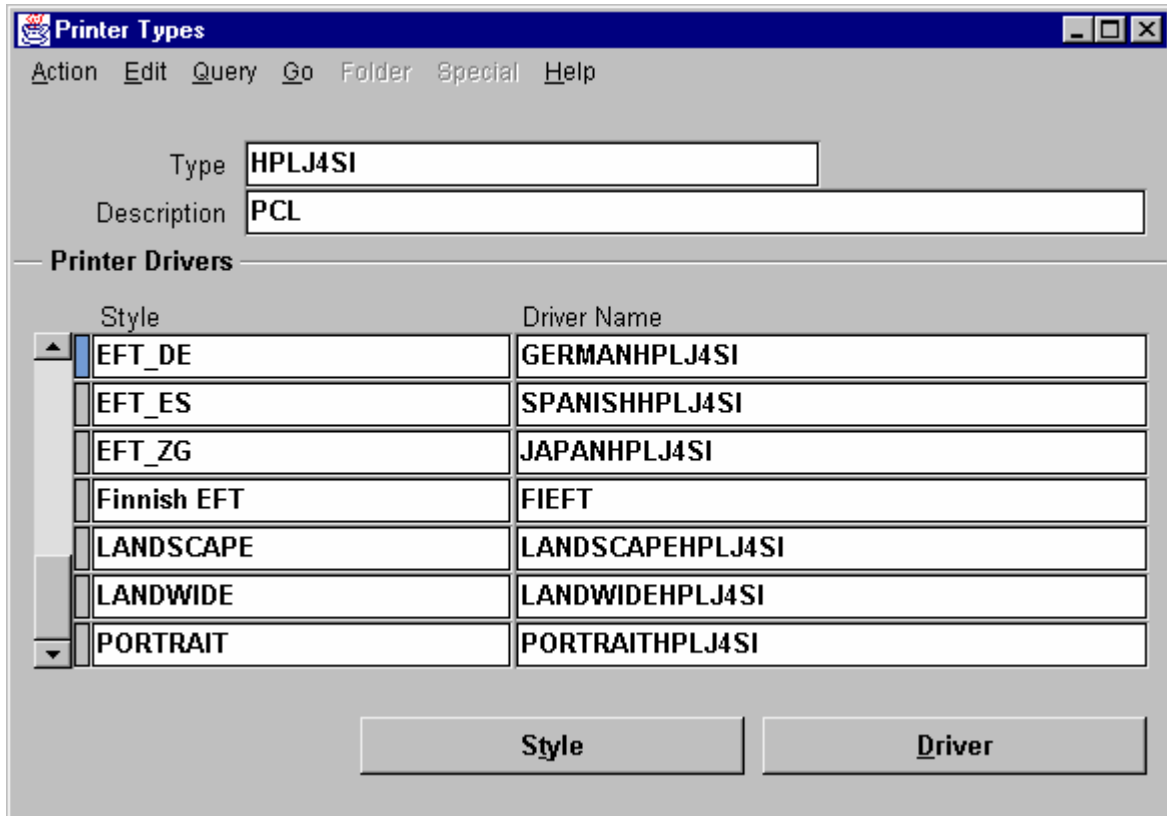


Fig 5 Sample of the printer type define window.

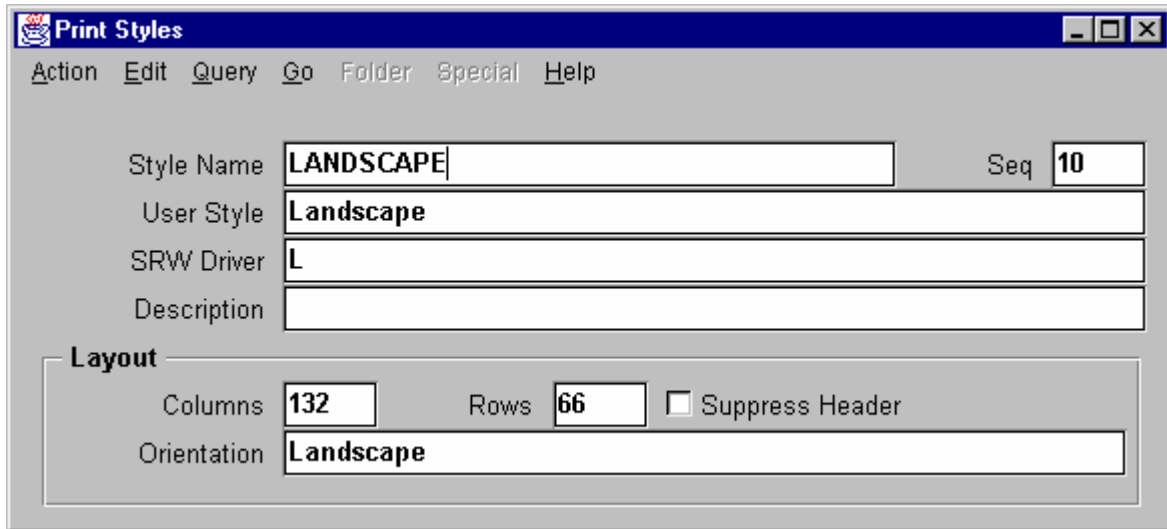


Fig 6 Sample of the print style define window.

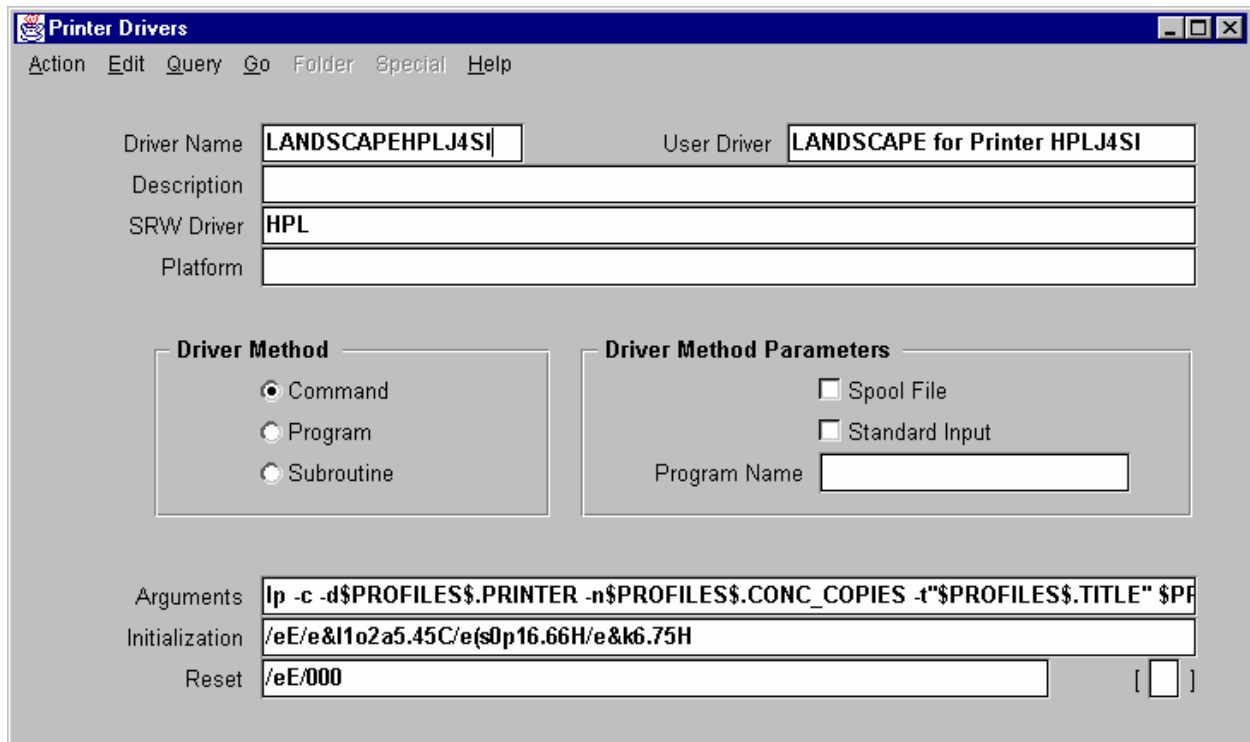


Fig 7 Sample of the define printer driver window.

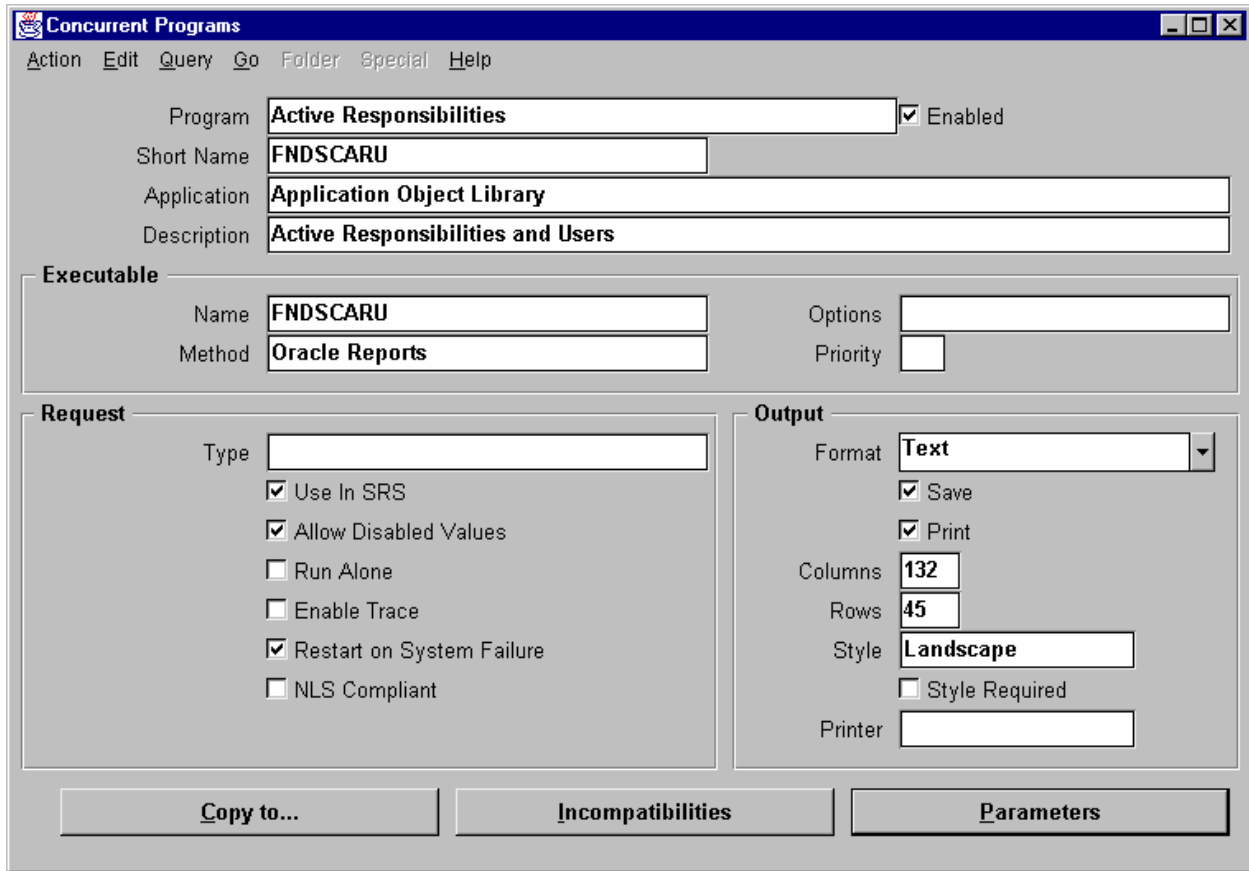


Fig 8 Sample of the register concurrent program window.

The screenshot shows the 'System Profile Values' window with a table of viewer profile options:

Profile	Site	Application	Responsibility	User
Viewer: Default Font Size	10			KGILLINS
Viewer: HTML				
Viewer: MIME Type for Text				application/ftview
Viewer: PDF				
Viewer: PostScript	c:\msoffice\winwor			
Viewer: Text	c:\windows\notepa			browser

Fig 9 Viewer profile options.

System Profile Values				
Action Edit Query Go Folder Special Help				
Profile	Site	Application	Responsibility	User
				KGILLINS
▲ RRA: Delete Temporary Files	No			
▣ RRA: Enabled	Yes			
▣ RRA: Maximum Transfer Size				
▣ RRA: Service Prefix				
▣ RRA: URL				
▣				
▣				
▣				
▣				
▣				
▣				
▣				
▣				

Fig 10 Report review agent profile values.

System Profile Values				
Action Edit Query Go Folder Special Help				
Profile	Site	Application	Responsibility	User
				KGILLINS
▲ Concurrent: Show Requests Su	No			
▣ Concurrent: Use ICM	No			
▣ Concurrent:Active Request Lim				
▣ Concurrent:Attach URL	Yes			
▣ Concurrent:Collect Request Sta				
▣ Concurrent:Conflicts Domain				
▣ Concurrent:Debug Flags				
▣ Concurrent:Directory for Copy				
▣ Concurrent:Hold Requests	No			
▣ Concurrent:Multiple Time Zone	No			

Fig 11 Concurrent processing profile values.

System Profile Values					
Action Edit Query Go Folder Special Help					
Profile	Site	Application	Responsibility	User	
Concurrent:Multiple Time Zone	No			KGILLINS	
Concurrent:Report Access Level	User				
Concurrent:Report Copies	0				
Concurrent:Request Priority	50				
Concurrent:Request Start Time					
Concurrent:Save Output	Yes				
Concurrent:Sequential Request	No				
Concurrent:Show Request Set					
Concurrent:URL Lifetime					
Concurrent:Wait for Available					

Fig 12 Concurrent processing profiles values.

```

$GTECOMM_TOP/$APPLBIN/storqm $PROFILES$.GTE_ALLOW_STORQM
$PROFILES$.REQUEST_ID $PROFILES$.FILENAME $PROFILES$.TITLE
"10KMTHTCT" "756"

```

Search OK Cancel

Fig 13 Arguments value for printer driver.

```

# FILE:storqm
#####
#   Written by Kevin Gillins      March 1997      #
#   This script is to generate and send reports to storqm.  The parameters  #
#   needed here will come from the printer driver assignments.            #
#   accounts.                                                                #
#####

### Display processing parameters to user in log file

sqm_allowed=$1
req_id=$2
source=$3
user_rep=$4
target=$5
rep_id=$6

req_log=$APPLCSF/$APPLLOG/l$req_id.req

echo =====
echo ===== >> $req_log
echo '      Allowed:'$sqm_allowed      >> $req_log
echo '      Request:'$req_id           >> $req_log
echo '      Report:'$user_rep          >> $req_log
echo '      Source:'$source            >> $req_log
echo '      Target:'$target            >> $req_log

if [ "X$sqm_allowed" != "XY" ]
then
    echo You are not allowed to send reports to STORQM >> $req_log
    echo Please contact the system administrator to have >> $req_log
    echo Your ALLOW_STORQM set to Yes. >> $req_log
    echo 'STORQM NOT ALLOWED'
    echo =====
    echo ===== >> $req_log
    exit 1
else
    echo Printing report $target to STORQM on `date`
    echo by $user_rep
    echo user ftp_user transrpt > $req_log.ftp
    echo put $source $target.$rep_id >> $req_log.ftp
    echo put $req_log.go $rep_id.GO >> $req_log.ftp
    echo close >> $req_log.ftp
    chmod 600 $req_log.ftp
    ftp -n rptserv < $req_log.ftp
    rm $req_log.ftp
    echo =====
    echo ===== >> $req_log
fi

exit 0

```

Fig 14 Sample shell script used for report archiving.