

Peregrine

ServiceCenter

Report Writer

Release 6

© Copyright 2004 Peregrine Systems, Inc.

PLEASE READ THE FOLLOWING MESSAGE CAREFULLY BEFORE INSTALLING AND USING THIS PRODUCT. THIS PRODUCT IS COPYRIGHTED PROPRIETARY MATERIAL OF PEREGRINE SYSTEMS, INC. ("PEREGRINE"). YOU ACKNOWLEDGE AND AGREE THAT YOUR USE OF THIS PRODUCT IS SUBJECT TO THE SOFTWARE LICENSE AGREEMENT BETWEEN YOU AND PEREGRINE. BY INSTALLING OR USING THIS PRODUCT, YOU INDICATE ACCEPTANCE OF AND AGREE TO BE BOUND BY THE TERMS AND CONDITIONS OF THE SOFTWARE LICENSE AGREEMENT BETWEEN YOU AND PEREGRINE. ANY INSTALLATION, USE, REPRODUCTION OR MODIFICATION OF THIS PRODUCT IN VIOLATION OF THE TERMS OF THE SOFTWARE LICENSE AGREEMENT BETWEEN YOU AND PEREGRINE IS EXPRESSLY PROHIBITED.

Information contained in this document is proprietary to Peregrine Systems, Incorporated, and may be used or disclosed only with written permission from Peregrine Systems, Inc. This book, or any part thereof, may not be reproduced without the prior written permission of Peregrine Systems, Inc. This document refers to numerous products by their trade names. In most, if not all, cases these designations are claimed as Trademarks or Registered Trademarks by their respective companies.

Peregrine Systems, AssetCenter, AssetCenter Web, BI Portal, Dashboard, Get-It, Get-Services, Get-Resources, Peregrine Mobile, and ServiceCenter are registered trademarks of Peregrine Systems, Inc. or its subsidiaries.

Microsoft, Windows, Windows NT, Windows 2000, SQL Server, and names of other Microsoft products referenced herein are trademarks or registered trademarks of Microsoft Corporation. Oracle is a registered trademark of Oracle Corporation. zlib software copyright © 1995-2004 Jean-loup Gailly and Mark Adler ServiceCenter includes cryptographic software written by Eric A. Young (eay@cryptsoft.com). ServiceCenter includes software written by Tim J. Hudson (tjh@cryptsoft.com). Spell Checker copyright © 1995-2004 Wintertree Software Inc. Smack software copyright © Jive Software, 1998-2004. SVG Viewer, Mozilla JavaScript-C (SpiderMonkey), and Rhino software Copyright © 1998-2004 The Mozilla Organization. OpenSSL software copyright (c) 1998-2004 The OpenSSL Project. All rights reserved. MX4J software copyright © 2001-2004 MX4J Team. All rights reserved. JFreeChart software © 2000-2004, Object Refinery Limited. All rights reserved. JDOM software copyright © 2000 Brett McLaughlin, Jason Hunter. All rights reserved. LDAP, OpenLDAP, Java, and the Netscape Directory SDK Copyright © 1995-2004 Sun Microsystems, Inc. Japanese Morphological Analyzer ©2004 Basis Technology Corp. The Sentry Spelling-Checker Engine Copyright © 2000 Wintertree Software Inc. CoolMenu software copyright © 2001 Thomas Brattli. All rights reserved. Coroutine Software for Java owned by Neva Object Technology, Inc. and is protected by US and international copyright law. Crystal Reports Pro and Crystal RTE software © 2001 Crystal Decisions, Inc., All rights reserved. Eclipse software (c) Copyright (c) 2000, 2004 IBM Corporation and others. All rights reserved. Copyright (C) 2001-2004 Kiran Kaja and Robert A. van Engelen, Genivia inc. All rights reserved. Xtree copyright 2004 Emil A. Eklund. This product includes software developed by the Indiana University Extreme! Lab (<<http://www.extreme.indiana.edu/>>).

The information in this document is subject to change without notice and does not represent a commitment on the part of Peregrine Systems, Inc. Contact Peregrine Systems, Inc., Customer Support to verify the date of the latest version of this document. The names of companies and individuals used in the sample database and in examples in the manuals are fictitious and are intended to illustrate the use of the software. Any resemblance to actual companies or individuals, whether past or present, is purely coincidental. If you need technical support for this product, or would like to request documentation for a product for which you are licensed, contact Peregrine Systems, Inc. Customer Support by email at support@peregrine.com. If you have comments or suggestions about this documentation, contact Peregrine Systems, Inc. Technical Publications by email at doc_comments@peregrine.com. This edition of the document applies to version 6.0 of the licensed program.

040507

Peregrine Systems, Inc.
3611 Valley Centre Drive San Diego, CA 92130
Tel 800.638.5231 or 858.481.5000
Fax 858.481.1751
www.peregrine.com



Contents

	Introducing Report Writer	5
	Knowledge requirements	5
	Examples	6
	Need further assistance?	6
	Customer Support	6
	Documentation Web site	6
	Education Services Web site	6
Chapter 1	Report Writer	7
	Overview	7
	Understanding Report Writer	8
	How to Access Report Writer	8
	How to Create a Simple Report Listing	9
	Creating and Editing Report Formats	21
	How to Query for Reports	22
	How to Use Query Formats with Reports	24
	Creating Primary File Queries	24
	How to Use Initial Formats with Reports.	30
	Accessing Forms Designer.	31
	How to Use Page Breaks and Report Totals.	32
	Accessing the Totals Sections of a Report	32
	Using Simple Subtotals	38
	Using Subtotals, Summary Totals, and Grand Totals	40
	Using Calculations in a Report.	43
	Accessing Format Control.	45

How to Use Format Control with Reports	47
Format Control and Report Formats	47
Using Format Control for Subheaders.	47
Detail Formats.	52
Using Format Control in a Matrix Report	53
Subtotal Format	55
Total Format	56
How to Run Reports in a Foreground	60
How to Schedule Reports for Background Processing	61
Understanding Background Report Processing	61
Scheduling a Report for Background Processing	62
Online Viewing of Reports.	65
Accessing the Spoolheader and Spool Files	66
How to View Report Schedule Records	70
How to Copy/Rename Reports	70
How to Delete Reports	71
How to Use Stacked Queries for Sorting	73
Primary Query Option	74
Stacked Queries Option.	74
Rules to Follow When Defining Stacked Queries	76
Report Writer Options Menu Selections	78

Introducing Report Writer

Report Writer is a component of the ServiceCenter toolkit. You can use it to create Report Writer reports and query for reports. With Report Writer, you can create report formats, which you can later update to add complexity or clarify details.

Knowledge requirements

Report Writer users need a working knowledge of their ServiceCenter system. System Administrators using this guide must have a working knowledge of ServiceCenter applications and utilities.

You can find more information in the following:

- For information about a particular platform, see the appropriate platform documentation.
- For database configuration information, refer to the *ServiceCenter Database Management and Administration Guide*.
- For information about using ServiceCenter, customizing your environment using parameters, administration and configuration, go to the *ServiceCenter* online help.
- For copies of the guides, download PDF versions from the CenterPoint web site using the Adobe Acrobat Reader, which is also available on the CenterPoint Web Site. You can also order printed copies of the documentation through your Peregrine Systems sales representative.

Examples

The sample windows and the examples included in this guide are for illustration only, and may differ from those at your site.

Need further assistance?

For further information and assistance with this release, you can download documentation or schedule training.

Customer Support

For further information and assistance, contact Peregrine Systems Customer Support at support.peregrine.com.

If the KnowledgeBase does not contain an article that addresses your concerns, you can search for information by product; search discussion forums; and search for product downloads.

Documentation Web site

For a complete listing of current ServiceCenter documentation, see the Documentation pages on the Peregrine Customer Support Web.

You can view PDF files, including release notes using Adobe Reader™, which is available at www.adobe.com.

Education Services Web site

Peregrine Systems offers classroom training anywhere in the world, as well as “at your desk” training via the Internet. For a complete listing of Peregrine’s training courses, see www.peregrine.com/education. You can also contact Peregrine Systems Education Services at +1 858.794.5009.

1 Report Writer

CHAPTER

Overview

This chapter describes Report Writer.

The following topics are discussed.

- *How to Access Report Writer* on page 8.
- *How to Create a Simple Report Listing* on page 9.
- *How to Query for Reports* on page 22.
- *How to Use Query Formats with Reports* on page 24.
- *How to Use Initial Formats with Reports* on page 30.
- *How to Use Page Breaks and Report Totals* on page 32.
- *How to Use Format Control with Reports* on page 47.
- *How to Run Reports in a Foreground* on page 60.
- *How to Schedule Reports for Background Processing* on page 61.
- *Online Viewing of Reports* on page 65.
- *How to View Report Schedule Records* on page 70.
- *How to Copy/Rename Reports* on page 70.
- *How to Delete Reports* on page 71.
- *How to Use Stacked Queries for Sorting* on page 73.

Understanding Report Writer

Report Writer allows you to define selection, totaling, and report format criteria to create listings from one or more database files, using one database file as the primary file. The reports may be as simple or complex as you wish, and can be modified at any time to add complexity or clarify details. The Report Writer allows for complete user-definition.

Note: The Report Writer is best viewed using the Web client.

How to Access Report Writer

- 1 Enter `rw` on any menu command line and press **Enter**.

The *report.prompt.rw* form opens.

- Search for an existing report or primary file name.

Press **Enter** or click **Search** to perform a *true* query without entering any values in the blank form and bring up a QBE list of existing report records. From the QBE list select a report to edit, delete, or run.

-or-

Enter an existing report or primary file name to be queried and press **Enter** or click **Search**. From the QBE list select a report to edit, delete, or run.

- Create a new report.

Click **Create New Report** to begin the process of creating a new report. The *report.maint* form opens. For detailed information on creating a new report, see *How to Create a Simple Report Listing* on page 9 for detailed information.

- 2 Click **Cancel** and then **Back** to exit Report Writer.

Option Keys

Listed below are the option keys available from the *report.maint* form.

Option Key	Description
F3 = OK	Ends the <i>edit</i> mode, saves the newly-created report or any modifications made to an existing report, and returns to the <i>format report</i> .
F4 = Cancel	Cancels out of creating a new report and does not save the information. Returns back to the <i>report.prompt.rw</i> format.
F5 = Run	Begins the process of running a report in the foreground or background. A newly-created report is saved and modifications made to an existing report are also saved.

After selecting **Edit** or when creating a new report, the following options are displayed:

Option	Description
Forms Designer	Allows you to access Forms Designer to alter existing or create new forms.
Set Query	Begins the process of defining Primary File Queries. Refer to <i>How to Use Query Formats with Reports</i> on page 24 for more information.
Totals	Opens a window at the bottom of the current screen, which allows for the definition of Total Parameters. Refer to <i>How to Use Page Breaks and Report Totals</i> on page 32 for more information.
Copy/Rename	Begins the process of copying or renaming a report. Refer to <i>How to Copy/Rename Reports</i> on page 70 for more information.

How to Create a Simple Report Listing

This section demonstrates how to create a simple report listing:

- 1 Enter `rw` on any menu command line and press **Enter**.
The *report.prompt.rw* form opens.
- 2 Enter the name of the new report in the **Report** field of the *report.prompt.rw* form, and then click the **Create New Report** button.

For this example, enter `problem.category` as the name of the report in the **Report** field.

Note: Choose a unique name for the report and enter it here. Any naming scheme can be used. It is recommended that you do not use special characters other than “.” or blank spaces. Usually the first node up to the first “.” is the same as the primary file of the report.

For this example, if you enter `problem category` without the “.” (period), the primary file is not created. If you enter `problem.category`, the primary file is created.

The system checks to see if the report format already exists.

- If the report format already exists, the following message appears in the status bar: *The report name <report name> is already used as a format. Please re-enter.*

You can either make changes to the existing report, or try entering a different name to create a new report.

- If the report format does not exist, a *report.maint* form opens (Figure 1-1).

The **Header** and **Details** fields in the Formats structure of the form are pre-filled with the name of the report and an added extension of *.header* and *.detail*, respectively.

The screenshot shows the 'Report Maintenance' dialog box in the Peregrine software. The dialog is titled 'Report Maintenance' and has a 'Basic' tab. It contains several sections:

- Name:** Fields for Name, Title, Application, Initial Form, Primary File, and Query Form.
- Author:** falcon
- Editor:** [empty]
- Date/Time:** 02/11/2004 15:34:35
- Save Report:**
- Format Type:** Radio buttons for Summary, Detail, Auto, and Manual.
- Formats:** Fields for Header, Subhdr, and Detail. Checkboxes for Print master Header, Print Subheader, and Subheader Page Break.
- Select:** Fields for Primary Query, Stacked Queries, and Sort Sequence. A table for Lines Per Page (66) and Width (80/132).
- Subtotals:** A table with columns for Field/Variable Names to Total, Calc, and Variable Names on Format.

Figure 1-1: Creating a new report format

- Using the Tab key to move the cursor, fill in the fields. For information on field descriptions, see *Field Descriptions for the report.maint Form* on page 12 and *Field Descriptions of the report.totals Form* on page 35.

Note: Be sure to fill in the *required* fields.

- When you are finished adding information to the necessary fields, click OK. You will receive a message in the status bar that states: *Record added to the report file.*

Note: The maximum width of a report is 132 characters. If the formats that are defined for a report exceed this limit, they are generated with that width to the `spool` file. However, when passed to the printer (Unix), they are truncated to 132 characters (or the maximum width as defined in the `config` file for the printer that is used to print the report).

Note: To set the width of a report, complete the **Page Width** field of the Client Printing structure in the user's Operator record. For more details on completing the Operator record, see the *Administering ServiceCenter* online help.

After the necessary fields have been defined, you must create the *header* and *detail* formats that are used by the report (if they have not already been created). For more information, see *Creating and Editing Report Formats* on page 21.

Field Descriptions for the *report.maint* Form

Field	Description
<i>Basic Structure</i>	
Name	Choose a unique name for the report and enter it here. Any naming scheme can be used; however, it is recommended that you do not use special characters other than "." or blank spaces. Usually the first node up to the first "." is the same as the primary file of the report. This field is required.
Title	Enter the report title that will print at the top of every page as part of the <i>master header</i> format.
Application	ServiceCenter application being used for this report.
Initial Form	Defines the name of the format to be used to obtain data from the operator at run time, which can then be used during report generation. Refer to <i>How to Use Initial Formats with Reports</i> on page 30 for more information.

Field	Description
Primary File	<p>Name of the primary ServiceCenter file that is used to select records, such as the probsummary file, to satisfy the query requirements of the report.</p> <p>This file is the logical file that uses the queries that are defined in the fields: Primary File, Query and/or Stacked Queries.</p> <p>If this field is left blank, it will default to being the same as the first node (up to the first “.”) of the Report Name after Create New Report is pressed to create a new report. If the report name does not contain a “.”) then it defaults to blank when Create New Report is pressed to create a new report. The next displayed format (<i>report.maint</i>) must then be defined. This field is required.</p> <p>For this example, this field is defaulted to problem, because a primary file was not specified on the <i>report.prompt.rw</i> format. The report name does have at least one “.”) (such as, <i>problem.category</i>).</p>
Query Form	<p>Defines the format to be used that will allow the operator to define via QBE searching techniques the primary file query to use to select records for the report. The QBE is translated to an SQL format and is stored by operator ID and report name.</p> <p>Any number of queries can be stored per report and they can then be accessed and used for a particular execution of a report. Refer to <i>How to Use Query Formats with Reports</i> on page 24 for more information.</p>
Author	This field is populated by ServiceCenter, which uses the operator’s login.
Editor	This field is populated by ServiceCenter, using the name of the last person (operator) who updated the report.
Save Report	The default is checked (<i>true</i>) to automatically save the new report.
Summary Detail	<p>Select one of the following valid entries:</p> <p>Summary = summary only prints subtotals, summary totals, and grand totals.</p> <p>Detail = details are selected from the Primary File, Query, or Stacked Queries and are printed using the format defined in the Detail field in the Format structure. Any other total/subheader formats are also printed.</p> <p>This field defaults to Detail.</p>

Field	Description
Auto	Select one of the following valid entries:
Manual	<p>Auto = If a Report Type was not defined on the <i>report.prompt.rw</i> format in the previous step, it is defaulted to auto.</p> <p>If a report is defined as auto, the standard functions of the Report Writer application (Page Breaks, Totals, Format Control) can be used during report generation.</p> <p>Manual = If a report is defined as manual, user-defined RAD panels are required to perform all functions of the report (such as, report selection, calculations, and page breaks).</p>
<i>Formats structure</i>	
Header	<p>The <i>header</i> format name is a user-defined format that is printed at every top-of-page break. This format will be printed in addition to the master header, and usually contains column headings that explain the fields in the detail section of the report.</p> <p>Note: If a <i>header</i> format is not defined for a report but Yes is specified for the Print master Header field and Detail is selected to print in the Summary/Detail field, both the <i>header</i> format and the <i>detail</i> format will print.</p> <p>Note: The maximum width of a report is 132 characters. If the header formats that are defined for a report exceed this limit, they are generated with that width to the spool file. However, when passed to the printer (Unix), they are truncated to 132 characters (or the maximum width as defined in the config file for the printer that is used to print the report).</p> <p>Note: To set the width of a report, complete the Page Width field of the Client Printing structure in the user's Operator record. For details on completing the Operator record, see online help.</p>

Field	Description
Subhdr	<p>Name of the user-defined format to print subheaders. The <i>subheader</i> format is used when a report is designed to print information from a file that immediately precedes detail information extracted from a secondary file. If subheaders are used, totals may be calculated only for the subheader records. Subheader reports require a Format Control record to be defined for the <i>detail</i> format of the report, which identifies the secondary file that is used to extract information for the <i>detail</i> format.</p>
Detail	<p>Name of a user-defined format, which is used for printing the detail information of each record extracted from the primary file or, if subheaders are used, it will print the detail information extracted from the secondary file.</p> <p>The field names referenced on this format must be field names from the primary or secondary file's database dictionary and/or variables that are defined through Format Control or Totals Parameters definitions.</p> <p>The naming standard for this field is the report name (problem.category) with the character string .detail appended (for example, problem.category.detail). However, this naming convention does not have to be followed. As a default, the detail format of the report takes on the name of the newly created report with an added extension of <i>.detail</i>.</p> <p>The same detail format can be used on multiple reports, if desired. This field is required.</p>

Field	Description
Print master Header	<p>Valid entries are:</p> <p>Yes = Yes, which is the default. If Yes is selected, one of two formats (<i>report.master.hdr132</i> or <i>report.master.hdr80</i>, depending on the value specified in the Width field) is printed every time an auto report takes a top-of-page exit.</p> <p>Note: The formats <i>report.master.hdr132</i> and <i>reportmaster.hdr80</i> print the following information:</p> <ul style="list-style-type: none"> ■ Company title as defined in the System Wide Company Record in the info file. ■ Date/time the report started processing. ■ Report title. ■ Page number. ■ Operator ID of operator who generated report. ■ Selection criteria (if query is <i>true</i>, then it is reported as <i>all records</i>, otherwise the actual query is printed). ■ Sort fields. <p>No = when only the format specified in the field header format name is printed during top-of-page exits. If No is specified and the field header format name is blank, only the format specified in the field detail format name is printed. The fields normally printed on a master header may be incorporated into the user-defined <i>header</i> format.</p> <p>Note: To set the width of a report, complete the Page Width field of the Client Printing structure in the user's Operator record. For more details on completing the Operator record, see online help.</p>
Print Subheader	<p>Yes = when a subheader is defined and Yes is selected, the subheader is printed on the report. The default used is Yes (see Figure 1-1 on page 11).</p> <p>No = when a subheader is defined and No is selected, the subheader is not printed on the report.</p>
Subheader Page Break	<p>Valid entries are:</p> <p>Yes = when a subheader is defined and Yes is selected, it is printed at top-of-page breaks. The default used is Yes.</p> <p>No = when a subheader is defined and No is selected, the subheader format is printed once for every detail record.</p> <p>Note: A subheader format has to be defined in order for this selection to have any significance on the report. For this example, a subheader is not used.</p>

Field	Description
<i>Select structure</i>	
Primary Query	<p>Defines either the query that is used to extract records from the primary file during processing of the report or if a query is used to extract records from a sortwork file (if one or more <i>stacked queries</i> are defined).</p> <p>At user discretion, the query can only be defined for a particular report, or can be used in conjunction with stacked queries. This field is required.</p> <p>Note: For performance considerations, you must have a good understanding of the relationship difference between the primary file query and stacked queries. Refer to <i>How to Use Stacked Queries for Sorting</i> on page 73 for more information.</p>
Stacked Queries	<p>An array (list) field that can optionally contain multiple queries that are used to extract records from the primary file. Refer to <i>How to Use Stacked Queries for Sorting</i> on page 73 for more information.</p>
Sort Sequence	<p>An array (list) field used to specify the sort sequence of the detail records extracted, using the primary file query or stacked queries.</p> <p>If a primary file query is defined and no stacked queries are defined, the sort fields <i>must</i> be defined as <i>keys</i> in the primary file's dbdict.</p> <p>If at least one stacked query is defined, the sort fields do not have to be defined as keys, but they <i>must</i> be defined as <i>fields</i> in the primary file's dbdict. Refer to <i>How to Use Stacked Queries for Sorting</i> on page 73 for more information.</p> <p>For this example, the report does not use stacked queries, therefore the <i>sort sequence</i> must be defined as a key in the primary filename (probsummary) dbdict. The sort sequence used for the demonstration report is <i>assignment</i>.</p>
Lines Per Page	<p>Defines how many lines per page the report allows, including the lines of the <i>master header</i> format, <i>header</i> format, <i>subheader</i> format, and <i>detail</i> format. The default is 66 lines per page.</p> <p>Users who direct their listings to laser printers may want to consider changing this value to 60, because of the 1/2 inch margin requirement for the bottom and top of a page. Otherwise, the page may overflow. This field is required.</p>

Field	Description
Width (80/132)	<p>Control which master header format is printed during top-of-page breaks.</p> <p>If the value is 80 or less, the master header format <i>report.master.hdr80</i> is used.</p> <p>If the value is greater than 80, the master header format <i>report.master.hdr132</i> is used. The default used is 80.</p> <p>This field does not perform any type of truncation on any formats used in the report. The value entered is not validated. This field is required.</p> <p>Note: To set the width of a report, complete the Page Width field of the Client Printing structure in the user's Operator record. For more details on completing the Operator record, see online help.</p> <p>For this example, report width of 80 is used.</p>
<i>Subtotals structure</i>	
Format Name	<p>The name of the format that is used to print accumulated statistics at a user-defined control/condition break. More than one statistic can be accumulated and printed for each control break.</p> <p>To add totals to the Subtotals structure, select Options > Totals.</p>
Field/Variable Names to Total	<p>A field name that is defined in the primary file's dbdict, or a variable that is defined in an earlier subtotals definition within the same report, or a variable that is defined in a Format Control definition for the <i>detail</i> format of the report.</p> <p>Report Writer will use the value of this field to perform the indicated calculation each time a <i>detail</i> record is processed.</p>
Calc	<p>One of four choices that indicates which type of calculation is performed using the value of the field or variable defined in field/variable names to total. The indicated calculation is performed for every detail record processed by the report regardless of whether or not the detail format is printed.</p>

Field	Description
SUM	<p>Accumulates the values of a specified field for each detail record extracted by the Primary File Query or Stacked Queries.</p> <p>If a variable is specified for the Field/Variable Name To Total field, then the value contained in that variable at the time each detail record is printed (or would have been printed if d was specified for summary or detail on the <i>report.maint</i> format) is accumulated in the variable specified in the Variable Names On Format field.</p> <p>The accumulation continues until some control/condition break is reached, at which time the accumulated total is stored in the variable defined in the Variable Names On Format field, the <i>subtotal format</i> is printed, and the variable used to accumulate the totals is reset to zero.</p>
AVG	<p>Accumulates the values of a specified field for each detail record extracted by the Primary File Query or Stacked Queries, and the total number of detail records read that produced the accumulated total.</p> <p>If a variable is specified for the Field/Variable Name To Total field, then the value contained in that variable at the time each detail record is printed (or would have been printed if d was specified for summary or detail on the <i>report.maint</i> format) is accumulated in the variable specified in the Variable Names On Format field.</p> <p>When a control/condition break is detected, the accumulated total is divided by the total number of records. This result is stored in the variable defined in the Variable Names On Format field.</p> <p>The control/condition break also causes the <i>subtotals format</i> to print, causing the variable used to accumulate the totals and the total number of records to be reset to zero.</p>

Field	Description
MIN	<p>Examines the value of a specified field in each detail records and retains the lowest value encountered.</p> <p>If a variable is specified for the Field/Variable Name To Total field, then the value contained in that variable at the time each detail record is printed (or would have been printed if d was specified for summary or detail on the <i>report.maint</i> format) is accumulated in the variable specified in the Variable Names On Format field.</p> <p>When a control/condition break is detected, the value is stored in the variable defined in the Variable Names On Format field.</p> <p>The control/condition break also causes the <i>subtotals format</i> to print, causing the variable used to accumulate the totals and the total number of records to be reset to zero.</p>
MAX	<p>Examines the value of a specified field in each detail record and retains the highest value encountered.</p> <p>When a control/condition break is detected, the value is stored in the variable defined in the Variable Names On Format field.</p> <p>The control/condition break also causes the <i>subtotals format</i> to print, causing the variable used to accumulate the totals to be reset to zero.</p>
Variable Names on Format	<p>Defines the variable that contains the results of the specified calculation. This variable is normally printed via the <i>subtotals format</i>. However, this is not a requirement. The variable can also be used in subsequent calculations.</p>

Creating and Editing Report Formats

The following steps help you to create and edit report formats directly from Report Writer.

- 1 Enter `rw` on any menu command line and press **Enter**.

The `report.prompt.rw` form opens.

- 2 Enter the name of the existing report in the **Report** field of the `report.prompt.rw` form, and then press **Enter** or click **Search**.

For this example, enter `problem.category` (the name of the newly created report discussed in [How to Create a Simple Report Listing](#) on page 9).

The `report` form opens (Figure 1-2 on page 21).

- 3 Click **Edit** to update the report and add any necessary information. The `report.maint` form opens (Figure 1-2). See the field descriptions discussed in the [Field Descriptions for the report.maint Form](#) on page 12.

The screenshot shows the 'Report Maintenance' form in Report Writer. The form is titled 'Report Maintenance' and is divided into several sections:

- Basic:**
 - Name: problem.category.report
 - Title: OPEN PROBLEM ANALYSIS REPORT - CATEGORY
 - Author: scott
 - Editor: scott
 - Application: problem
 - Initial Form: Summary
 - Primary File: problem
 - Query Form: Detail
- Formats:**
 - Header: problem.category.header
 - Subheader: problem.category.detail
 - Detail: problem.category.detail
 - Print Master Header: Yes
 - Print Subheader: [Empty]
 - Subheader Page Break: [Empty]
- Select:**
 - Primary Query: header,status~="closed" and header,last=true
 - Stacked Queries: [Empty]
 - Sort Sequence: header,category
 - Lines Per Page: 66
 - Width (80/132): 80
- Subtotals:**

Format Name	Field/Variable Names to Total	Calculation	Variable Names on Format

Figure 1-2: Editing a report in the `report.maint` form

- 4 If the report format name has not been previously entered, select **Options > Forms Designer**. At the **enter format name** prompt, enter the *detail format* name.

For this example, enter `problem.category.detail` and press **Enter**.

Fill in the detail format, following normal Forms Designer procedures. Select **Options > Forms Designer**. For information on Forms Designer, see the *Administering ServiceCenter* online help.

Note: When filling in a *detail* format for a report, only input fields are defined. However, this is not a restriction. Label fields can also be defined, if needed.

- 5 Press **F1** to add.

Note: Edit the field names to reflect their dbdict field names. To ensure that the input field names are correct, the format may be associated with the **primary** file of the report. As each input field is defined, check the field name to verify that the field does exist in the **primary** file.

- 6 Press **F3** to close the edit field window.

- 7 Press **F3** until you return to the *report.maint* format and save the newly-created format.

Note: Update the *header* format, if prompted to do so.

When you are ready to run this report, refer to *How to Run Reports in a Foreground* on page 60 and *How to Schedule Reports for Background Processing* on page 61 for complete instructions.

How to Query for Reports

- 1 Enter `rw` on any menu command line and press **Enter**.

The *report.prompt.rw* form opens.

- 2 Press **Enter** or click **Search** without entering any values to perform a *true* query and pull up a QBE list of existing report records.

-or-

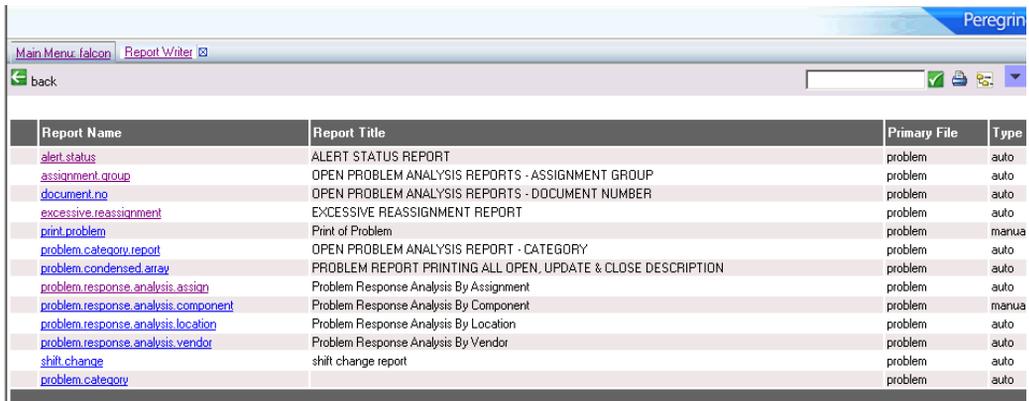
Enter an existing report name to be queried and press **Enter** or click **Search**.

- 3 For this example, query for any report that has a **primary** file of `problem` or `probsummary`. Enter `problem` in the **Primary File** field of the *report.prompt.rw* form. Press **Enter**.

If the query entered selects more than one record, a QBE list of reports appears (Figure 1-3).

Note: Beginning with ServiceCenter 5.0, the primary file to query for problem records is *probsummary*. Use the *problem* file when the Use **Paging** field is checked in the **Incident Management Environment** record.

If the query entered produces only one record, that report record opens on the *report* screen.



Report Name	Report Title	Primary File	Type
alert_status	ALERT STATUS REPORT	problem	auto
assignment_group	OPEN PROBLEM ANALYSIS REPORTS - ASSIGNMENT GROUP	problem	auto
document_no	OPEN PROBLEM ANALYSIS REPORTS - DOCUMENT NUMBER	problem	auto
excessive_reassignment	EXCESSIVE REASSIGNMENT REPORT	problem	auto
print_problem	Print of Problem	problem	manua
problem_category.report	OPEN PROBLEM ANALYSIS REPORT - CATEGORY	problem	auto
problem_condensed_array	PROBLEM REPORT PRINTING ALL OPEN, UPDATE & CLOSE DESCRIPTION	problem	auto
problem_response_analysis_assign	Problem Response Analysis By Assignment	problem	auto
problem_response_analysis_component	Problem Response Analysis By Component	problem	manua
problem_response_analysis_location	Problem Response Analysis By Location	problem	auto
problem_response_analysis_vendor	Problem Response Analysis By Vendor	problem	auto
shift_change	shift change report	problem	auto
problem.category		problem	auto

Figure 1-3: A QBE list of reports displays in response to the query

- 4 Scroll through the list. Place the cursor on any record within the QBE list and double-click on that record to select the record. The record opens on the *report* form (see Figure 1-2 on page 21).

The report can be edited, deleted, or scheduled to run. When you are ready to run this report, see *How to Run Reports in a Foreground* on page 60 and *How to Schedule Reports for Background Processing* on page 61.

- 5 Click **Back** or press **F3** to return to the main Report Writer menu (*report.prompt.rw* form).

How to Use Query Formats with Reports

Report Writer provides the option to define a set of pre-defined queries for a report, which can then be used as the *primary file query* when the report is initially scheduled to run. These queries are stored by report name and user ID, and can be accessed for viewing and updating.

Creating Primary File Queries

When creating primary file queries, you have two options:

- Enter the name of the form to be used when creating queries in the **Query Form** field.
- or-
- Leave the **Query Form** field blank, which causes the Report Writer to use the value in the **Primary File** field for the name of the format to be used in creating queries.

Note: The query form name must be valid in order to be used.

To access a report for viewing and updating when a record is selected:

- 1 Click **Edit** or press F2.

The *report.maint* form appears (Figure 1-2 on page 21). This allows you to edit the current report.

- 2 Click **OK** to save your changes.

A message displays in the status bar that states: *Record updated in the report file.*

-or-

Click **Run** to begin running the report.

A message appears in the status bar that states: *Record updated in the report file.*

The Run Report Exerciser (*report.prompt.exer*) form opens and you can fill in specifics on this form to run the report.

To learn more about running reports, see *How to Run Reports in a Foreground* on page 60 and *How to Schedule Reports for Background Processing* on page 61.

To establish pre-defined queries for a report:

- 1 Enter `rw` on any menu command line and press **Enter**.
The `report.prompt.rw` form opens.
- 2 Press **Enter** or click **Search** to display a QBE list of existing report records.
-or-
Enter an existing report name to be queried.
For this example, enter `problem.category.report`. The `report` form opens (Figure 1-2 on page 21).
- 3 Click **Edit** or press **F2** (edit report) to access the edit functions of Report Writer.
- 4 Select **Options > Set Query** to access the query format and begin the process of setting queries for the report (Figure 1-4 on page 25). When working with this format, use the option keys described in the following section, *Option Keys* on page 25.

Figure 1-4: Report Query Maintenance

Option Keys

Listed below are the Option keys available from the `report.query.maint` form.

Option Key	Description
F1 = add	Adds the currently displayed query and name to the query database. The name field must be unique per user and report.
F2 = update	Applies changes made to the query and/or name.
F3 = back	Returns to the previously displayed format.
F4 = delete	Deletes the currently displayed query record.

Option Key	Description
F5 = clear	Clears all displayed fields.
F6 = build	Displays the format defined in the Query Form . If no format is specified, the format that is the same name as the <i>primary filename</i> of the report (if any) appears.
F8 = update rpt qry	Copies the currently displayed query and overlays the <i>primary file query</i> of the report currently being edited.

For this example, select the *problem.category.report* record.

The Report Query Maintenance (*report.query.maint*) form displays.

- If report queries were already defined for this report, a QBE list of those queries opens. From this list, you can select a query for maintenance.
- If there are no report queries defined for this report, a message appears in the status bar that states: *No stored queries exist for you for this report. Press F6 to add.*

- 5 Select a record from this list and click **Edit** or press **F2** to make changes.

When you finish editing the record, the following message appears in the status bar: *Report query record has been updated.*

-or-

Enter the name of the query in the **Name** field and the query into the **Query** field of the *report.query.maint* form.

For this example, enter the following information into the *report.query.maint* form.

- a Enter **Alert Status** in the **Name** field.
- b Enter the following in the **Query** field:

status#"alert" or status#"DEADLINE"

- 6 Click **Add** to add the stored query.

A message appears in the status bar that states: *Report query record has been added.*

- 7 To begin the process of entering a report query, press **F6** or click **Build**.

The *problem* form opens (Figure 1-5). The format that you specify here when you enter QBE arguments appears in the **Query** field of the *report.query.maint* form (Figure 1-4 on page 25).

Figure 1-5: Entering a report query

- 8 Enter data in the fields that will satisfy your QBE search. The more fields you fill in, the narrower your QBE search will be. The fields are described at the end of these steps in *Field Descriptions for the problem Form* on page 29
- For this example, enter the field value `=systems` in the **Category** field.
- 9 Press **F6** (query window) to open the query form. This form displays the SQL translation of the query entered on the screen, which can be modified.

-or-

Press **Enter** to translate the query into SQL and the *report.query.maint* format is redisplayed with that SQL expression in the **Query** field.

For this example, press **F6**.

Note: This routine does not automatically insert the `flag=true` portion of queries, which are normally associated with queries used against Incident Management and Change Management.

The user is responsible for making the necessary modification to the translated SQL to accommodate these or any other special query needs.

For this example, press **Enter**.

The screenshot shows a web-based application window titled 'Peregrine'. The window has a menu bar with 'Main Menu: falcon' and 'Report Writer'. Below the menu bar is a toolbar with buttons for 'Back', 'Add', 'Update', 'Delete', 'Clear', 'Build', and 'Update Rpt Qty'. The main content area is titled 'Report Query Maintenance'. It contains two input fields: 'Name:' with the value 'Alert Status' and 'Query:' with the value 'header,category#"systems"'.

Figure 1-6: Completed Report Query Maintenance build

- 10 If the query is correct, press **Enter**.

-or-

If the query shown in the *report.query.maint* form is not the desired query, make changes as needed and click **Update**.

For this example, the desired query is shown, which will select all records from the *probsummary* file that have a *category* value of *systems*.

The translated SQL appears in the **Query** field.

- 11 Define a unique query name for the query.
For this example, enter *systems*.
- 12 Optionally, you can modify the SQL expression shown in the **Query** field.
For this example, add **flag=true** to the existing expression in the **Query** field.
- 13 Press **F1** or click **Add** to add the new query record. You will receive a message displayed at the bottom of the screen, stating: *Report query has been added*.
- 14 Click **Back** to get back to the *report.maint* form.
- 15 Click **Run** to run the newly created report query.
A QBE list of one or more queries opens.
- 16 Double-click the query name of one of the query records listed.
- 17 The Run Report Exerciser (*report.prompt.exer*) form opens, from which you can schedule the report to be run in the background or foreground.

For more information on processing reports, see *How to Schedule Reports for Background Processing* on page 61 and *How to Run Reports in a Foreground* on page 60.

While in the *report.prompt.exer* form, the query that was defined in the previous steps now appears in the **Primary File Query** field. However, the original primary file query defined in the report record itself is *not* affected.

Figure 1-7: Processing reports using the Run Report Exerciser

Field Descriptions for the *problem* Form

The field descriptions in the *problem* form describe how you will set up your data to satisfy your QBE search. The Header Information structure delineates how each record is printed within the query.

Field	Description
<i>Header Information structure</i>	
Incident No	Incident number of queried records.
Status	Status of the call reports to be used for this query.
Category	Category by which an incident has been classified.

Field	Description
Assigned To	Qualified person to which the incident has been assigned.
Priority	Priority (1 through 4) assigned to the call report.
Change No.	Unique change record number.
Page	Page number of the printed report.
Reassign Count	Query incident tickets that have been reassigned a specified number of times.
Opened	Date the incident ticket was opened.
Alert	If an alert has been generated for this record, the alert status that was generated.
Updated	Date the record was last updated.
Closed	Date the record was closed.
Logical Name	Logical file name for this record.
Location	Query a particular company location to be selected, or leave blank to include all company locations.
Contact Name	Query for a particular contact name.
Service Contact	Query for a particular service contact -- a contact outside of business hours.

How to Use Initial Formats with Reports

An *initial format* allows the user to enter data, which assists in developing the query for the report.

For example: If the report requires a date and time range, it is simpler to provide the user with a format to enter this range rather than entering a complete SQL type query in the **primary file query** field.

If an *initial format* is required to retrieve data from the user at runtime, the format name must be entered in the **initial format** field in the *report.maint* form.

Accessing Forms Designer

To start Forms Designer:

For detailed information on using Forms Designer, see the [Administering ServiceCenter](#) online help.

- 1 When in the *report.maint* or *report.totals* form, go to **Options > Forms Designer**.

-or-

Click Toolkit located on the system administrator's home menu.

The selections available in the Toolkit appear. Click the **Forms Designer** button.

The initial Forms Designer view appears.

A Drawing Canvas and Design Tools open. When you begin to create the new report form, the Properties view displays the properties for the selected control.

With the Drawing Canvas, Design Tools, and Properties view you can begin to design and create a new report.

For detailed information on creating a new report in Forms Designer, see the [Administering ServiceCenter](#) online help for the usual Forms Designer procedures.

For this example, we will access the *problem.category* report.

- 2 From the System Administrator's menu, enter *rw* on any menu command line and press **Enter** or click **Search**.
- 3 Enter *problem.category* in the **Report** field of the *report.prompt.rw* form and click **Search**.

The *report* form opens (Figure 1-2 on page 21).

- 4 Click **Edit** or press **F2** (edit report) to access the edit functions of Report Writer.

The *report.maint* form opens (Figure 1-2 on page 21). See the [Field Descriptions for the report.maint Form](#) on page 12.

- 5 Go to **Options > Forms Designer** to create the format, using the usual Forms Designer procedures. For more information on Forms Designer, see the [Administering ServiceCenter](#) online help.

Note: **\$start** and **\$end** are the input field names for the fields shown when creating your new form in Forms Designer.

- 6 To include the information that will be entered in the initial format, change the primary file query statement to:

number>\$start and number<\$end

- 7 Press F5 to run the report.

Note: Reports using the *initial format* option cannot be run in the background. Therefore, the **Run report in background** field is defaulted to f (*false*). See *How to Schedule Reports for Background Processing* on page 61 for more information.

- 8 Press F1 to complete the request.

See *How to Run Reports in a Foreground* on page 60 and *How to Schedule Reports for Background Processing* on page 61 for more information about running reports.

How to Use Page Breaks and Report Totals

In addition to producing simple report listings, Report Writer also allows for accumulation of various statistics, based on user-specified totals and parameters. These statistics can be printed, according to user-defined formats at various phases of report generation based on user-defined control/condition breaks. Variations of Totals Parameters can be used with or without Format Control definitions, which provide the ability to create a wide variety of report statistics.

Accessing the Totals Sections of a Report

This section demonstrates accessing the Totals section of a report and defines the various Totals Parameters, including how to reference them and how to create the formats that are used to print accumulated statistics.

To access the Totals section of a report:

For this example, we will access the *problem.category* report.

- 1 Enter `rw` on any menu command line and press **Enter**.
The *report.prompt.rw* form opens.
- 2 Enter `problem.category` in the **Report** field.
The *report* form opens.
- 3 Click **Edit** or press F2 (edit report) to access the edit functions of Report Writer.

The *report.maint* form opens (Figure 1-2 on page 21). See the *Field Descriptions for the report.maint Form* on page 12.

4 Select **Options > Totals** to access the *report.totals* form.

This is a lengthy form, so use the scroll bars to move to the left and to the right, as well as up and down (see the form pictured in two sections, Figure 1-8 on page 33 and Figure 1-9 on page 34). See the *Field Descriptions of the report.totals Form* on page 35.

Report Totals

Basic

Report Name: problem.category.report

Title: OPEN PROBLEM ANALYSIS

Primary File: problem

Select

Primary Query: header.category#"systems"

Stacked Queries:

Subtotals

Format Name	Field/Variable Names to Total	Calc	Variable Names on Format

Figure 1-8: First section of the *report.totals* format

Reset Field/Variable Name	ALWAYS	Summary	Subtotal Expression
<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>
<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>
<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>

Figure 1-9: Second section of the report.totals format

Field Descriptions of the *report.totals* Form

Field	Description
<i>Totals Structure</i>	
Format Name	Name of the format used to print accumulated statistics at a user-defined control/condition break. More than one statistic can be accumulated and printed for each control break. To add totals to the Subtotals structure, select Options > Totals .
Field/Variable Names To Total	Either defined in the primary file's dbdict, defined as a variable in an earlier subtotals' definition within the same report, or defined as a variable in a Format Control definition for the <i>detail format</i> of the report.
Calc	One of four choices that indicates which type of calculation is performed, using the value of the field or variable defined in field/variable names to total . Note: The indicated calculation is performed for every detail record processed by the report, regardless of whether or not the detail format is printed. Valid Calc (calculation) types are: <p>SUM Accumulates the values of a specified field for each detail record extracted by the Primary File Query or Stacked Queries. If a variable is specified for the Field/Variable Name To Total field, then the value contained in that variable at the time each detail record is printed (or would have been printed if d was specified for summary or detail on the <i>report.maint</i> format) is accumulated in the variable specified in the Variable Names On Format field.</p> <p>The accumulation continues until some control/condition break is reached, at which time the accumulated total is stored in the variable defined in the Variable Names On Format field, the <i>subtotal format</i> is printed, and the variable used to accumulate the totals is reset to zero.</p>

Field	Description
AVG	<p>Accumulates the values of a specified field for each detail record extracted by the Primary File Query or Stacked Queries, and the total number of detail records read that produced the accumulated total.</p> <p>If a variable is specified for the Field/Variable Name To Total field, then the value contained in that variable at the time each detail record is printed (or would have been printed if d was specified for summary or detail on the <i>report.maint</i> format) is accumulated in the variable specified in the Variable Names On Format field.</p> <p>When a control/condition break is detected, the accumulated total is divided by the total number of records. This result is stored in the variable defined in the Variable Names On Format field.</p> <p>The control/condition break also causes the <i>subtotals format</i> to print, causing the variable used to accumulate the totals and the total number of records to be reset to zero</p>
MIN	<p>Examines the value of a specified field in each detail records and retains the lowest value encountered.</p> <p>If a variable is specified for the Field/Variable Name To Total field, then the value contained in that variable at the time each detail record is printed (or would have been printed if d was specified for summary or detail on the <i>report.maint</i> format) is accumulated in the variable specified in the Variable Names On Format field.</p> <p>When a control/condition break is detected, the value is stored in the variable defined in the Variable Names On Format field.</p> <p>The control/condition break also causes the <i>subtotals format</i> to print, causing the variable used to accumulate the totals and the total number of records to be reset to zero</p>
MAX	<p>Examines the value of a specified field in each detail record and retains the highest value encountered.</p> <p>When a control/condition break is detected, the value is stored in the variable defined in the Variable Names On Format field.</p> <p>The control/condition break also causes the <i>subtotals format</i> to print, causing the variable used to accumulate the totals to be reset to zero.</p>
Variable Names on Format	<p>Defines the variable that contains the results of the specified calculation. This variable is normally printed via the <i>subtotals format</i>. However, this is not a requirement. The variable can also be used in subsequent calculations.</p>

Field	Description
New Page	Indicates whether or not Report Writer should print the next detail record at the top of a new page <i>after</i> the indicated <i>subtotals format</i> has been printed. Valid entries are: <i>true</i> (yes), <i>false</i> (no), or <i>blank</i> (<i>false</i> and blank are synonymous). The default is blank. This can also be used to define a page break condition with no subtotals being printed.
Reset Field/Variable Name	The field name or variable name that is used to test for control/condition breaks. Every time the value of the field/variable changes, the indicated <i>subtotals format</i> is printed and all variables defined in Variable Names On Format field are reset to zero and the next detail record is printed after the report takes a top-of-page break. If no <i>subtotals format</i> is defined, only a top-of-page break is taken on the condition break.
ALWAYS	If <i>true</i> is specified, the <i>subtotals format</i> is printed every time a control break is detected (even if only one detail record has been subtotaled). If <i>false</i> (or blank) is specified, the <i>subtotal format</i> is <i>not</i> printed if only one detail record has been processed when a control/condition break is detected. The default is blank.
Summary	If <i>true</i> is specified, the totals defined will print in the summary format. If <i>false</i> (or blank) is specified, the <i>summary format</i> will not print. Enter true for all entries defining Summary Totals. Leave blank for all other types of totals.
Subtotal Expression	An expression that is evaluated against every detail record processed to further qualify when the indicated calculations will be performed. If the expression evaluated to <i>true</i> against the current detail record, the indicated calculations are performed. Note: The results of subtotals expressions are printed at the end of the report after all subtotals and before any summary totals.

Using Simple Subtotals

The following sample report demonstrates how to define the necessary totals parameters to cause Report Writer to print a *subtotals format*.

For this example, we will use the demonstration report, *problem.category.report*, created in *How to Create a Simple Report Listing* on page 9.

This report generates subtotals from the Primary file (**problem**), showing the total number of records for each Assignment Group.

To access the totals section within this report, see the instructions discussed in *Accessing the Totals Sections of a Report* on page 32. See the steps described below to create a totals format.

To create a Totals format in Forms Designer:

- 1 Enter the name of the format being used to print the subtotal in the **Format Name** field of the Totals structure (see Figure 1-8 on page 33).

For this example, enter `probsummary.demo.totals` and press **Enter**.

The system checks for a format called *probsummary.demo.totals* in the format file.

If the format already exists, it opens.

If the format does not exist, go to **Options > Forms Designer** to create a format using normal Forms Designer procedures. This format can be designed as you wish. For more information on Forms Designer procedures, see *ServiceCenter System Tailoring*.

- 2 Create each field, using the input properties described in *Input Properties Field Descriptions for the subtotals Form* on page 39.
- 3 Click **OK** to save your new form before exiting the Forms Designer to save the changes you made to the format.
- 4 Follow any other details described in the Forms Designer procedures discussed in *ServiceCenter System Tailoring*.
- 5 Scroll to the left to enter parameters in other fields on this format.

Note: Any time you scroll left or right while in this format, place the cursor in the home position before scrolling. This ensures that the labels and input fields remain properly aligned within the structured array.

- 6 To define the field by which you will be subtotaling, enter its name in the **reset/field variable name** field.

This causes the *subtotals format* (in this example, *problem.category.totals*) to print every time the value of the **assignment** field changes.

For example: If the **assignment** field value of the current detail record is **network control** and the next detail record has an **assignment** field value of **software support**, the subtotals are printed before the next detail record.

- 7 Scroll to the left again to enter parameters in the third section of this format.
8 Enter **true** in the **Always** field.

This forces the subtotals to print even if only one detail record is printed.

- 9 Press F3 (end) to return to the *report.maint* form.

Note: If you would like to erase all values entered in the *report.totals format*, press F4 (clear fields).

A confirmation prompt will display, allowing you to either press F1 (clear fields) to continue clearing the fields or press F3 (abort) to cancel deleting the *totals* information.

- 10 When you are ready to run this report, refer to *How to Run Reports in a Foreground* on page 60 and *How to Schedule Reports for Background Processing* on page 61 for complete instructions.

Input Properties Field Descriptions for the *subtotals* Form

The following input properties field descriptions are pre-defined by Report Writer. They are available for use when creating the *subtotals format* for any report you create in Forms Designer. For detailed information about Forms Designer procedures, see the *Administering ServiceCenter* online help.

Field	Description
report.count	Contains the total number of detail records processed by the report and is normally printed on a <i>grand totals format</i> , but may be printed on any totals format to show a running total.
report.subcount	An array that contains the total number of records processed for each calculation of each subtotal. If it is necessary to reference a particular array element on a <i>subtotals format</i> , the correct field name would be report.subcount,x where x is an array element number relative to the position of a calculation within the totals definition. This field is normally used in conjunction with the report.sumflds field for summary totals.

Field	Description
<code>report.sumflds</code>	An array that contains the value of each field that caused a control break. This field is normally used in conjunction with <code>report.subcount</code> for summary totals.
<code>report.subfld</code>	Contains the value of the field entered in the reset field/variable name at the time a control/condition break is detected.
<code>\$rw.dummy</code>	A variable defined by Report Writer that contains <i>NULL</i> characters.
<code>\$rw.months</code>	An array that contains full month names. <i>January</i> is the first element with <i>December</i> being the twelfth element.
<code>\$rw.mon</code>	An array that contains abbreviated (three-character) month names. <i>Jan</i> is the first array element with <i>Dec</i> being the twelfth element.
<code>\$rpt.date</code>	A variable defined by Report Writer that contains the date and time when the report was executed.
<code>\$page</code>	A variable defined by Report Writer that contains the page of the output of the report.

Using Subtotals, Summary Totals, and Grand Totals

This report demonstrates how to define the necessary Totals Parameters to cause Report Writer to print subtotals, summary totals, and grand totals.

Summary Totals

Display the same statistics that were printed in each subtotal, but in a summarized format. Changing the **summary/detail** field on the *report. maint format* to *s* will cause only the totals portions (subtotals, summary totals, and grand totals) of the report to print. The *detail format* will not print.

Grand Totals

Show the total number of detail records that were selected during report generation. This report will use the `enduser inventory` file as the *primary file*.

Create the report through Report Writer. See *How to Create a Simple Report Listing* on page 9 for detailed instructions on creating a report, including creating the outlined header and detail formats procedures.

- 1 Use the following parameters to create this report.
 - a Enter a unique name in the **Report Name** field.
For this example, enter **enduser.count**.
 - b Enter any user-defined title in the **Report Title** field.
For this example, enter **Monthly Inventory Count of Enduser Device by Model**.
 - c Select **Auto** in the **Auto/Manual** field.
 - d Select **Detail** in the **Summary/Detail** field.
 - e Enter **enduser** in the **Primary File name** field.
 - f Use the default value of **enduser.count.header** for the **Header Format Name** field.
 - g Use the default value of **enduser.count.detail** in the **Detail Format Name** field.
 - h Enter **true** in the **Primary File Query** field.
 - i Enter **model** in the **Sequence of Report** field.
 - j Use the following fields for the *detail format*:
 - model
 - logical.name
 - serial.no.
 - location
- 2 When you are finished adding the report, press F2 (edit rpt) to access the edit functions.
The report opens.
- 3 Press F4 (totals) to access the *report.totals format*.
- 4 Enter the name of the format being used to print the subtotal in the **Totals Format Name** field.
The *subtotals format* can be created through the normal Forms Designer procedures either before entering the Totals Parameters or during the Totals Parameters definition process. For more information on Forms Designer procedures, see the *Administering ServiceCenter* online help.

For this example, create the subtotal format **enduser.count.subtot** from the *report.totals* form.

- 5 Position the cursor on the **Totals Format Name** field and go to **Options > Forms Designer**.

The system checks for a format called **enduser.count.subtot** in the format file. If that format already exists, it opens.

If the format does not exist, a blank screen will display, allowing you to fill in the format using the normal Forms Designer procedures.

For information on Forms Designer procedures, see the *Administering ServiceCenter* online help.

The format may be designed as needed.

Note: The value in parentheses () (**\$rw.dummy**) is not intended to be a label on the format.

- 6 Edit each field to update the format. Be sure to update the format when prompted.
- 7 Click **OK** to return to the *report.totals* form.
- 8 To force a page break after each subtotal format prints, enter **true** in the **new page** field.
- 9 To access the second portion of the *report.totals* form, press **Home** to move the cursor from the array, and scroll to the left until you see the **label reset/field variable name**.

To define the field by which you will be subtotalling, enter its name in the **reset/field variable name** field. This causes the subtotals format (enduser.count.subtot) to print every time the value of the field **model** changes.

For this example, enter **model**.

- 10 Scroll to the left again to enter parameters in the third section of this format.

Using Calculations in a Report

Create the report through Report Writer. For detailed instructions on creating a report, including creating the outlined header and detail formats procedures, see *How to Create a Simple Report Listing* on page 9.

- 1 Use the following parameters to create this report.
 - a Type a unique name in the **Report Name** field.
For this example, enter `problem.category2`.
 - b Type any user-defined title in the **Report Title** field.
For this example, enter **Demo of How to Use Totals Calculations**.
 - c Type `auto` in the **Report Type** field.
 - d Select **Detail** in the **Summary/Detail** field.
 - e Type `probsummary` in the **Primary File name** field.
 - f Use the default value of `problem.category2.header` for the **Header Format Name** field.
 - g User the default value of `problem.category2.detail` in the **Detail Format Name** field.
 - h Type the following in the **Stacked Query** field: `flag=true`.
Note: Unix users can also enter `status~=“closed”`.
 - i Type `assignment` in the **Sequence of Report** field.
The system checks for a format called `enduser.count.total` in the format file.
If the format already exists, it opens.
If the format does not exist, a blank screen opens, allowing you to fill in the format using normal Forms Designer procedures.
For detailed Forms Designer procedures, see the *Administering ServiceCenter* online help. The format can be designed as needed.
 - j Use the following fields for the *detail format*:
number
category
assignment
status
open.time
\$demo.elapsed

- k When you are finished adding the report, press F2 (edit rpt) to access the edit functions. The report opens.
- 2 Press F4 (totals) to access the report.totals format.
- 3 Type the name of the format being used to print the subtotal in the **totals format name** field.

For this example, enter `problem.category2.subtotal`.

If the subtotals format needs to be created, go to **Options > Forms Designer**. For additional information on accessing Forms Designer, refer to *Accessing Forms Designer* on page 31. For detailed procedures on working within the Forms Designer utility, see *ServiceCenter System Tailoring*.

For this example, create the `problem.category2.subtotal`.

If the format does not exist, a blank screen is presented (see Figure 1-2 on page 21). Fill in the detail format, following normal Forms Designer procedures. Select **Options > Forms Designer**. For information on Forms Designer, see the *Administering ServiceCenter* online help.

Note: When filling in a *detail* format for a report, only input fields are defined. However, this is not a restriction. Label fields can also be defined, if needed.

- 4 To calculate the *average* and *sum* of the variable `$demo.elapsed` for each grouping of problem records by category, enter the values shown for the **field/variable names to total** and **calc** fields.
- 5 Scroll left to view the second portion of the *report.totals* format.
The values entered in the **variable names on format** array identify where the result of the calculation will be printed within the subtotal format.
Type the variable names used on the *problem.category2.subtotal* format.
- 6 Type the value category in the **reset field/variable name** field to calculate the totals separately for each category grouping.
- 7 When you are finished entering the appropriate data, press F3 (end) to return to the *report.maint* screen.

Using Format Control to Evaluate the Value to be Used for the \$demo.elapsed Variable

Format Control records can be attached to any form or file within ServiceCenter and do not require special programming skills to implement. Routines defined in Format Control can be user interactive or transparent, and are performed when a record opens or when a user adds, updates, or deletes a record from the database. Format Control is easily applied and changed.

The calculations process within Format Control can be used whenever you want to perform a calculation on currently available fields or variables. To evaluate the value to be used for the \$demo.elapsed variable used in the *problem.category2.detail* format, you must attach Format Control to the detail format.

Accessing Format Control

The best method for accessing Format Control is through Forms Designer. This method reduces the chances of accessing the incorrect record, because you are accessing the record directly from the form with the same name as the Format Control record.

To access Format Control:

- 1 When in the *report.maint* or *report.totals* form, go to **Options > Forms Designer**.

The initial Forms Designer view opens.

- 2 Enter the name of the form you wish to see in the **Form** field of the Forms Designer dialog box.

-or-

Click **Search** to find the form.

- 3 Go to **Options > Format Control**.

The Format Control record appears for the selected form.

Important: Be sure that the name of this Format Control record is the same name as your *detail format*.

Option Keys to be Used While in Format Control

Option Key	Description
F1 = add	To add the Format Control record for the <i>detail</i> format.
F2 = update	To update a Format Control record.
F3 = end	To exit the Format Control record when you are finished entering the calculation expressions. Continue to press F3 to exit and return to the <i>report.maint</i> form.
F5 = calcs	Accesses the calculations section of the Format Control record. Example of a calculation expression: if (flag=false) then (\$demo.elapsed=close.time in \$file - open.time in \$file) else (\$demo.elapsed=tod() - open.time in \$file)
F6 = more	Accesses other sections of the Format Control record.
F12 = end (no add)	Cancels out of creating a new report and does not save the information. Returns back to the <i>report.prompt.rw</i> format.

When you're ready to run the report, refer to *How to Run Reports in a Foreground* on page 60 and *How to Schedule Reports for Background Processing* on page 61.

How to Use Format Control with Reports

When a report is created using Report Writer, these reports use several forms to generate the report. There are the *Initial*, *Header*, *Detail*, *Subheader*, and *Totals* forms. Each of these forms can have Format Control records set up to be used by Report Writer when reports are executed at runtime of the report.

Format Control and Report Formats

Report Format	Description
Initial Formats	Format Control on the <i>initial</i> format can be set up to initialize data entered on the <i>initial</i> format, using the initialization expressions section to validate data entered on the <i>initial</i> format. Use the F1 (add) option of any section. The Format Control record would look like Format Control on any database format.
Header Formats	Format Control on the <i>header</i> format can be set up to initialize variables that can be used by the other formats and are defined in the initialization expressions section of the Format Control record. If a variable is to be initialized only once while the report is being executed, it should be done in the <i>header</i> format.
Detail Formats	Format Control on the detail format can be set up to initialize or calculate data, select records from secondary files, and execute P4 subroutines, using the display option of the Format Control record. Validation expressions will have no effect in this version of the Report Writer. If a variable is set in the <i>detail</i> format, it will be reset each time a record is selected.

Using Format Control for Subheaders

The *subheader* format is used when a report is designed to print information from a file that immediately precedes detail information extracted from a secondary file.

Subheader reports require a Format Control record to be defined for the *subheader* format of the report, because the *subheader* format identifies the secondary file that is used to extract information for the *detail* format.

The *subheader* format contains fields from the primary file defined in the report. The *detail* format contains fields from the secondary file, which is defined using the Secondary File Queries section of the Format Control record of the *subheader* format.

The primary and secondary files are connected by a common field value.

For detailed information on using Format Control, refer to *ServiceCenter System Tailoring*.

In this example, we will match the value in the **location** field from the **location** file (primary file) with the value in the **location** field from the **enduser** file (secondary file). Each time the report selects a record based on the primary file query from the primary file (**location** file), Report Writer uses Format Control to find the matching records in the secondary file (**enduser** file) using the **location** field.

To create a report using Format Control:

- 1 Create a new report in Report Writer (see Figure 1-10 on page 49). For details on creating a new report, refer to *How to Create a Simple Report Listing* on page 9.

Use the following values when creating the report in the *report.maint* form.

Field	Value
Name	location.summary (or any unique name)
Title	A user-defined title
Auto or Manual	Auto.
Summary or Detail	Detail
Primary File	location
Header Format	location.summary.header (default)
Subhdr Format	location.summary.subhdr
Detail Format	location.summary.detail (default)

Field	Value
Primary Query	true
Sort Sequence	location

Note: location must be a key in the primary file's dbdict before it can be used as a sort field, since stacked queries are not used.

The screenshot shows the 'Report Maintenance' dialog box in the Peregrine software. The dialog is titled 'Report Maintenance' and is divided into several sections:

- Basic:** Name: location.summary, Title: User Defined, Application: , Initial Form: , Primary File: location, Query Form: . Author: falcon, Editor: , 02/12/2004 13:29:32. There are radio buttons for Save Report, Summary, Detail, Auto, and Manual.
- Formats:** Header: location.summary.header, Subhdr: location.summary.subhdr, Detail: location.summary.detail. Print master Header: Yes, Print Subheader: Yes, Subheader Page Break: .
- Select:** Primary Query: true, Stacked Queries: , Sort Sequence: location, Lines Per Page: 66, Width (80/132): 80.
- Subtotals:** A table with columns for Format Name, Field/Variable Names to Total, Calc, and Variable Names on Format.

Figure 1-10: Creating a new report

- 2 Click **Options > Forms Designer** to create the format in Forms Designer. A blank form opens.

A Drawing Canvas and Design Tools appear. When you begin to create the new report form, the Properties view also appears.

- 3 With the Drawing Canvas, Design Tools, and Properties view you can begin to design and create a new form for the report.

For detailed information on creating a new form for your report in Forms Designer, see the *Administering ServiceCenter* online help for the usual Forms Designer procedures.

- 4 When you are finished working within the Forms Designer utility, click **OK**. The new form appears in the `location.summary` (*forms.designerformat*) form. (If the format already exists, it opens.)
- 5 If you need to make further design changes, click **Design**.
- 6 When you are finished with your design changes, click **Options > Format Control** to create a new Format Control record.
- 7 You are prompted to save your changes. Click **Yes**.
The *formatctrl.maint.initial* form opens (Figure 1-11 on page 50).

Important: Make sure the name of your Format Control record is the same name as your *subheader* format.

In this example, there are no entries required in the **Initialization Expressions** field.

Figure 1-11: Format Control Maintenance form

- 8 Click **Add**.

The following message appears in the status bar: *Format Control record added.*

- 9 Click Options > Queries.
- 10 Click Yes.

The *formatctrl.maint.queries.b* form opens (Figure 1-12 on page 51).

Add	Upd Query	Del	Dis	Req'd	Init Message	Filename	Field Name to Check
<input type="text"/>							
<input type="text"/>							
<input type="text"/>							
<input type="text"/>							
<input type="text"/>							

Figure 1-12: Format Control Maintenance - File Queries form

- 11 Make the file queries entries as needed.

Field Value Entries

Field	Description
Dis	Enter true in the Dis (display) field.
Filename	Enter enduser . This identifies the enduser file as the secondary file for this report. The location file and enduser file will be connected when the values in their location fields match.
Query	Enter location=location in \$file .
Req'd	Enter false in the Req'd (required) field. This signifies that the match is not required for the subheader format to print.

- 12 Click **Save** to save the record.
- 13 Click **OK** twice to exit the forms and return to the *report.maint* form.

Detail Formats

To create detail formats:

For information on creating a report in Report Writer, refer to *How to Create a Simple Report Listing* on page 9.

For this example, we will use the *location.listing* report.

- 1 From the *report.maint* form, put your cursor in the **Detail** field of the Formats structure of the *report.maint* form (on the *location.summary.detail* value) and go to **Options > Forms Designer**.
The system checks for a format called *location.summary.detail* in the format file. If the format already exists, it opens. If it does not exist, a blank screen is presented, which allows you to paint the format using normal Forms Designer procedures.
For information on accessing Forms Designer or using Forms Designer, see the *Administering ServiceCenter* online help.
- 2 Click **Design** to create the format or change the existing format.
The *location.detail* design form opens.
The input fields for each field within the Forms Designer palette to be noted in the Properties - Text dialog box are as follows.

Input Fields

■ location.name	■ phone
■ location	■ crt.phone
■ address	■ loc.open.time
■ city	■ loc.close.time
■ state	■ other.contact
■ zip	■ other.contact.phone
■ primary.contact	

Note: The value in the **Input** field is the name of that **Input** field and is *not* intended to be a label on the format. You *must* enter a name into the **Caption** field of the Properties - Text dialog box in order to have the name of the field appear on the report.

- 3 Click **Add** or **OK** to return to the *report.maint* form.

Using Format Control in a Matrix Report

The following sample report gives summary count information on severity codes and the status of incident tickets. The severity code labels print in the column and the status (such as, open and update) labels print in a row.

The report consists of a *header* and two *totals* formats. It is a summary report, but it does not use the summary field (true/false) on the totals screen. There is a Format Control record associated with both the *header* and *detail* formats. They contain initialization and calculation expressions for processing record counts.

To create a matrix report using Format Control:

- 1 For details on creating a new report, refer to *How to Create a Simple Report Listing* on page 9.

Use the following values when creating the report in the *report.maint* form.

Field	Value
Name	problem.severity.summ (or any unique name)
Title	A user-defined title

Field	Value
Auto or Manual	Auto.
Summary or Detail	Summary
Primary File	probsummary
Header Format	problem.severity.summ.header (default)
Detail Format	problem.severity.summ.detail (default)
Primary Query	flag=true Note: Unix users can also enter status~="closed".
Sort Sequence	severity Note: In order to sort and break on the severity field for a summary type report, the same field must also be set up as a <i>key</i> in the probsummary dbdict, or the primary query could be defined as a stacked query. For more information on the Database Dictionary (dbdict), refer to the <i>Administering ServiceCenter</i> online help.

- When you have all the field values entered, place the cursor in the **Header** field (on the `problem.severity.summ.header` value) and click **Options > Forms Designer**.

The system checks for a format called `problem.severity.summ.header` in the format file. If the format already exists, it opens. If it does not exist, a blank screen opens, which allows the user to paint the format using normal Forms Designer procedures.

For detailed information on the Forms Designer procedures, refer to the *Administering ServiceCenter* online help.

Figure 1-13: Blank `problem.severity.summ.header` form

Note: In order to sort and break on the severity field for a summary type report, the same field must also be set up as a key in the probsummary dbdict, or the primary query could be defined as a stacked query.

- Click **OK** to add the new format and return to the `report.maint` form.
- To create a *detail* form, place your cursor in the **Detail** field (on the `problem.severity.summ.detail` value) and click **Options > Forms Designer**.

The system checks for a format called *problem.severity.summ.detail* in the format file. If the format already exists, it opens. If it does not exist, a blank screen is presented, which allows the user to paint the format using normal Forms Designer procedures.

Note: Since this report is a summary type report, the *detail* format will not print when the report is generated. However, because a Format Control record is required for calculations to occur on each detail record, we must create a *detail* format to attach to the Format Control record.

Subtotal Format

To create the subtotal format:

- 1 Position the cursor on the first Subtotals **Format Name** field and click **Options > Forms Designer**.

The initial Forms Designer (*format.prompt.initial*) form is opened.

- 2 Enter *problem.severity.summ.sub* and click **New**.
- 3 A question appears that asks if you want to use the Forms Designer wizard. Click **No**.

A blank Forms Designer palette (*problem.severity.summ.sub* form) is opened.

- 4 Follow normal Forms Designer procedures to create a new form. For information on Forms Designer procedures, refer to the [Administering ServiceCenter](#) online help.

Note: The value in parentheses in the Properties - Text dialog box is the name of the **Input** field and is not intended to be a label on the format. You must enter a caption in the **Caption** field in order to have the field name displayed on the report.

- 5 When you are finished, click **OK**.
- 6 From the **Options** menu, select **Totals**.

- 7 Fill in the required fields in the *report.totals* format as follows:

Subtotals Format Name	Field/Variable Names to Total
problem.severity.summ.sub	\$opn
	\$upd
	\$al1
	\$al2
	\$al3

- 8 Access the second portion of the *report.totals* format and scroll to the right until you see the **Reset Field/Variable Name** field.
Enter severity.
- 9 Enter true in the ALWAYS field.
Entering true causes the subtotals to print even if only one detail record is selected before the subtotal is reset.
- 10 Click OK.

Total Format

To create the total format:

- 1 Position the cursor on the second totals **Format Name** field and enter `problem.severity.summ.tot` and click **Options > Forms Designer**.
A blank Forms Designer view opens.
- 2 Follow normal Forms Designer procedures to create a new form. For information on Forms Designer procedures, refer to *ServiceCenter System Tailoring*.
Note: The value in parentheses in the Properties - Text dialog box is the name of the **Input** field and is not intended to be a label on the format. You must enter a caption in the **Caption** field in order to have the field name displayed on the report.
- 3 When you are finished with editing, click OK.
- 4 From the **Options** menu, select **Totals**.

- 5 Fill in the required fields in the *report.totals* format as follows:

Totals Format Name	Field/Variable Names to Total
problem.severity.summ.tot	\$opn
	\$upd
	\$al1
	\$al2
	\$al3

- 6 Scroll down and fill in the Variable Names on the Format field as follows.

Totals Variable Names on Format

\$open

\$updt

\$alrt1

\$alrt2

\$alrt3

\$opn.tot

\$upd.tot

\$alrt1.tot

\$alrt2.tot

\$alrt3.tot

- 7 Scroll over until you see the ALWAYS field.
Enter true.
- 8 When you are finished, click OK.

Format Control Record for the Header format

To create a Format Control record for the header format:

- 1 Position the cursor on the Format Header field and click **Options > Forms Designer**.
- 2 Create the necessary fields using the normal Forms Designer procedures.
For detailed information on the Forms Designer procedures, see the *Administering ServiceCenter* online help.
- 3 Click **OK**.
- 4 Click **Options > Format Control**.
- 5 When prompted, click **Yes**.

This will take you into the Format Control utility and allow you to add a new Format Control record (see Figure 1-11 on page 50).

Note: Be sure that the name of this Format Control record is the same name as your *header* format.

- 6 Enter the following statements in the **Initializations Expressions** field.

Initializations Expressions Values

```
$open=0 ; $updt=0 ; $alrt1=0 ; $alrt2=0 ; $alrt3=0
```

```
$opn.tot=0 ; $upd.tot=0 ; $alrt1.tot=0 ; $alrt2.tot=0 ; $alrt3.tot=0
```

Note: These initialization expressions are executed only once during the report generation.

- 7 When you are finished, click **OK**. Continue to click **OK** until you return to the *report.maint* form.

Format Control Record for the Detail format

To create a Format Control record for the detail format:

- 1 Position the cursor on the Format Detail field and click **Options > Forms Designer**.
- 2 Create the necessary fields using the normal Forms Designer procedures.
For detailed information on the Forms Designer procedures, see the *Administering ServiceCenter* online help.
- 3 Click **OK**.

- 4 Click **Options > Format Control**.
- 5 When prompted, click **Yes**.
This will take you into the Format Control utility and allow you to add a new Format Control record (see Figure 1-11 on page 50).
Note: Be sure that the name of this Format Control record is the same name as your *detail* format.
- 6 Enter the following statements in the **Initializations Expressions** field.

Initializations Expressions Values

```
$open=0 ; $updt=0 ; $all=0 ; $al2=0 ; $al3=0
```

Note: These initialization expressions will be executed each time a new *detail* record is selected.

- 7 When you are finished, click **OK**.
- 8 Click **Calculations** in the Format Control Maintenance form.
The following Format Control Maintenance - Calculations form opens.
- 9 Enter the values in the fields as described below.

Field	Values to be entered
display	true
calculation (enter all on one line)	If status in \$file#"ope" enter: \$opn=1
	If status in \$file#"upd" enter: \$upd=1
	If (index("1",status in \$file)>0) enter: \$al1=1
	If (index("2",status in \$file)>0) enter: \$al2=1

Field	Values to be entered
	If (index("3",status in \$file)>0) enter: \$al3=1
	Note: To increase performance, the above expressions should be spread out on one expression line (separated by semicolons).
	Important: <i>Never</i> place a semicolon at the end of the final expression. This causes the system to expect another expression.

- 10 When you are finished, click OK. Continue to click OK until you return to the *report.maint* form.

How to Run Reports in a Foreground

Running a report in the foreground dedicates the user's session and terminal to the task until the report has completed.

To set up reports to run in the foreground:

- 1 Access the report you wish to run. (For details on accessing a report, see *How to Access Report Writer* on page 8.)

The *report.maintenance* form opens (see Figure 1-2 on page 21).

- 2 When you have access to the *report* form, check the information to make sure it is correct.

Click **Edit** or press F2 to make any necessary modifications.

- 3 Click **Run** or press F5 to run the report.
- 4 From the *report.prompt.exer* form, you have the opportunity to modify the **summary or detail** and **primary file query** definitions.
 - a The **Print report after spooling** field defaults to true (yes). Change this to false (no).
 - b The **Run report in background** field defaults to true (yes). Change this to false (no).

If you leave the default set to *true*, the report will run at a set time in the background processor. To learn more about running reports in the background, see *How to Schedule Reports for Background Processing* on page 61.

If you change the default setting to *false*, the report is scheduled to pass to the printer and print immediately after it is generated.

After the report is finished spooling, the following message appears in the status bar: *Report spooled as no: xxx (report title)*.

- c The *report* form opens.
 - Press F1 to view the report. The spoolheader format opens, which allows you to view the report output online. For more information about viewing reports, see *Online Viewing of Reports* on page 65.
 - or-
 - Press F3 to end the task.

How to Schedule Reports for Background Processing

Understanding Background Report Processing

Running a report in the background means that a simulated user session within the ServiceCenter system is used to generate the report. The default background processor is named *report* and is normally made active, along with other background processors every time ServiceCenter is started, and is the processor that is used to process the report. The *info* file contains the definitions for all background processors and also defines which of those processors are automatically started at startup time.

When a report is scheduled to run in the background, information about the report is first added to the *schedule* file. The report background processor scans the *schedule* file every 60 seconds, which is the default scan interval, for any records whose scheduled time has elapsed and whose class is defined as *report*. When a record is found, the background processor performs all functions necessary to generate the report to the *spool* and *spoolheader* files. Reports may be scheduled to run in the future and be given a repeat interval to automatically reschedule. For example, you can schedule a report to run once every few hours, once a day, or once a week.

The *spoolheader* file contains information about the run statistics of the report (such as, the time the report started processing, the time the report finished, the total number of pages printed, and which operator scheduled the report for processing). The *spoolheader* file is also linked to the *spool* file.

The spool file contains the actual pages of the report. You can access the **spoolheader** file and view the generated reports online. To learn more about viewing reports online, refer to *Online Viewing of Reports* on page 65.

Normally after a report has been generated to the **spoolheader** and **spool** files, it is passed to the Unix systems based on parameters defined by the user at the time the report was scheduled to run.

Scheduling a Report for Background Processing

To schedule a report for background processing, access the report you wish to run in the background processor. To schedule a report for background processing:

- 1 Access a report, following the procedures discussed in *How to Access Report Writer* on page 8.

In this example, the *Problem Response Analysis by Assignment* report is opened in the *report.maint* form (see Figure 1-2 on page 21).

- 2 From the *report.maint* form, ensure that all the information about the report is correct.

If any information needs to be modified, click **Edit** or press **F2** and make the necessary changes.

- 3 When you are finished editing the report, click **Run** or press **F5** to schedule the report to be processed in the background.

The Run Report Exerciser (*report.prompt.exer*) form opens (see Figure 1-7 on page 29).

- 4 Click **Summary** to print a summary report.

-or-

Click **Detail** to print a detailed report.

- 5 If necessary, modify the **primary file query** field definitions.

To learn more about query formats within reports, refer to *How to Use Query Formats with Reports* on page 24.

- 6 Select one of the following print options.

- Select (check) **Print** to print the report when it is finished spooling. The report will print immediately after it has been generated.

Otherwise, leave the field blank (unchecked) to only spool the report. The report will remain in the spool file until it is manually deleted or scheduled to print.

-or-

- Select (check) **Background** to print the report in the background when it is finished spooling. The report is scheduled to run at a particular time (defined in the next screen sequence) in a background processor.

Otherwise, leave the field blank (unchecked) to print the report in the user's session.

To learn more about running reports in a user's session, refer to *How to Run Reports in a Foreground* on page 60.

- 7 Enter the *number* of lines per page in the **Lines Per Page** field that you would like printed on the report. Otherwise, it will default to the original settings entered in the *report* form.
- 8 Click **Run**.

The Schedule a Report (*report.prompt.sch2*) form opens.

The screenshot shows a web browser window titled 'Peregrine' with a tab for 'Report Exerciser'. The browser address bar shows 'Back', 'schedule', and 'Fill'. The main content area is titled 'Schedule a Report' and contains the following information:

*The time specified below indicates when the report will begin spooling.
The report will be scheduled for printing as soon as it has finished spooling.*

Time to Print:

Repeat Interval

Specific Int: Days HH:MM:SS (7 04:00:00 is 7 days, 4 hours)

OR place an "x" in one of the choices below

Monthly: Quarterly: Semi-Annually: Annually:

Printer

Name:

Class:

Defaults to config record setting if left blank

Controls

Number of Copies:

Delete Report:

Dist Group:

Figure 1-14: Scheduling a report to print

- 9 Schedule the report to print by entering the values in the fields of the *report.prompt.sch2* form. The fields are described below.

Field Descriptions

Field	Description
Time to Print	The date and time that the report will begin processing in the report background processor. It is defaulted to the current date and time, but can be changed as needed.
<i>Repeat Interval structure</i>	
Specific Int	<p>Defines a specific interval to establish a periodic frequency for when the report will be automatically rescheduled for processing.</p> <p>This is a relative date and time field and is added to the Time to Print field value each time the report is generated by the background scheduler.</p> <p>Note: If this field is used, <i>do not</i> select a frequency within the Monthly, Quarterly, Semi-Annually, or Annually options listed below.</p>
Monthly Quarterly Semi-Annually Annually	<p>Indicates the frequency in which the report should be run. Type X to select one of the following options: Monthly, Quarterly, Semi-Annually, or Annually.</p> <p>Note: Select only <i>one</i> option, otherwise you will receive a message stating that you must make only one selection.</p> <p>Note: If this field is used, <i>do not</i> use the Specific Int field.</p>
<i>Printer structure</i>	
Name	<p>Defines the printer to be used when printing the report. This is defaulted to being the same printer value as defined in the operator record for the user login name running the report. This can be changed to any valid printer on the network.</p> <p>To obtain a valid list of all the available printers, put your cursor in the Name field and click Browse. Double-click a printer in the Device Name field to select it from the list.</p>
<i>Controls structure</i>	

Field	Description
Number of Copies	The total number of copies of the report to be printed. The default number is one copy, but this number can be increased as needed.
Delete Report	Specify whether or not the report is automatically deleted from the spool and spoolheader files after it has been printed. If set to <i>true</i> (yes), the report is deleted. If set to <i>false</i> (no), the report will be saved in the spool and spoolheader files for viewing or rescheduling after it has been printed. The default setting is <i>true</i> (yes).
Dist Group	Defaults to <i>blank</i> . You can optionally click Browse to select a Group Name in the QBE list. Only one group can be selected as the distribution group. If a group is specified, all operators defined in the distribution group will receive the number of copies specified in the Number of Copies field. The copies for all operators will print to the designated printer.

- 10 After you have entered all the necessary values to generate the report, click **Schedule**.

You are returned to the Report Maintenance (*report*) form and the following message appears in the status bar: *Approx. time of exec: (00:00:00) for report: (name of report)*.

A record has been added to the schedule file that now contains all the pertinent run information.

- 11 Press **Enter** to erase the message and return to Ready.

On the Options menu, click **View Spooled Reports** to view the report. To learn more about viewing reports, refer to [Online Viewing of Reports](#) on page 65.

Online Viewing of Reports

When a report is generated in the foreground or in the background processor, it is added to the **spoolheader** and **spool** files to be printed or available for online viewing. The contents of the report cannot be modified, but the **spoolheader** file information can be adjusted, if necessary.

The spoolheader and spool files are both used to view reports. The **spoolheader** file defines such things as the name of the report, the time the report started spooling and ended spooling, and the operator who scheduled the report to be processed. The **spool** file contains the actual report (each page being a separate record within the spool file). The spoolheader and spool files are then linked to each other.

Accessing the Spoolheader and Spool Files

To view a report online:

- 1 Access a report, following the procedures discussed in *How to Access Report Writer* on page 8. In this example, the *resolution code listing* report opens in the *spoolheader* form.

The fields in the *spoolheader* form are described in *Spoolheader Form Fields* on page 67.

The screenshot shows a web application window titled 'Peregrine'. The main menu is 'falcon'. The application has a toolbar with buttons for 'View Report', 'Update', 'End', 'Delete', 'Sched Info', 'Refresh', 'Find', and 'Menu'. The main content area is titled 'SPOOLHEADER' and contains the following data:

Report			
Number:	887	Delete After Printing:	false
Pages:	1	Copies:	1
Name:	resolution code listing		
Status:	scheduled		
Messages:			
Spool			
Printer:	sysprint	Start Time:	02/12/2004 15:19:53
Pages Spooled:	1	End Time:	02/12/2004 15:19:53
Operator Name:	falcon		
Schedule			
Schedule Number:	887	Priority:	
Forms Class:		Orig Print Time:	02/12/2004 15:19:34
Printer Class:		Next Print Time:	
Print			
Printer:		Start Time:	
Pages Printed:		End Time:	

Figure 1-15: Accessing a report in the spoolheader file

Spoolheader Form Fields

Field	Description
<i>Report structure</i>	
Number	The unique report number assigned by the ServiceCenter system.
Pages	The number of pages generated in the report (the total number of times the report issued a top-of-page break).
Copies	The total number of report copies that will be printed (defined when the report was scheduled for background processing).
Name	The report title, as defined in the Report Writer record.
Status	<p>The current status of the report. This value is controlled by the ServiceCenter system. The following is a list of valid report statuses:</p> <ul style="list-style-type: none"> spooling - The report is current being processed in a user's session or in a background processor. spooled - The report is finished spooling and is ready for online viewing. scheduled - The report output has been scheduled to print. printing - The report is currently printing. printed - The report has completed printing and the delete report after printing field setting was <i>false</i> when the report was scheduled to print. (The report was set <i>not</i> to be deleted.) <p>For more information on the values to be set when scheduling a report to print, see How to Schedule Reports for Background Processing on page 61.</p>
<i>Messages</i>	
<i>Spool structure</i>	
Printer	Indicates which internal configuration definition is being used to obtain device information when the report is set to print.
Pages Spooled	Indicates the total number of pages generated by the report. If the report is currently being generated in the background, this number will increment until the report is finished processing.

Field	Description
Operator Name	<p>The operator who initiated the report for background processing or who actually ran the report in the user session.</p> <p>When the spoolheader form is initially displayed, this is defaulted to the operator ID of the current operator. If you leave the system default operator ID and press Enter, the system will search the spoolheader file for all records that have been generated by that operator ID. You can modify this field to search for any valid operator ID, or blank out the field altogether.</p> <p>Note: If you attempt to search the <i>spoolheader</i> records based on the default operator ID and no records are found, the system will return a message that states: <i>no records found</i>. The operator field will be blanked out.</p>
Start Time	The date and time that the report began processing in the foreground or background processor.
End Time	The date and time that the report ended processing in the foreground or background processor.
<i>Schedule structure</i>	
Schedule Number	After a report is scheduled to run, this field is filled in with a unique schedule file number associated with the report.
Forms Class	Any data filled in will override the <i>forms</i> value specified in the <i>config</i> record.
Printer Class	Any data filled in will override the <i>printer class</i> value specified in the <i>config</i> record.
Priority	<i>Not currently being used.</i>
Orig Print Time	The date and time that the report is scheduled to being spooling.
Next Print Time	When the report is run in the foreground and the print time is scheduled to run at a specific date with a repeat interval specified, the Next Print Time field reflects the scheduled date and the repeat interval.
<i>Print structure</i>	
Printer	Indicates the printer class to which the report is being generated.

Field	Description
Pages Printed	Indicates the total number of pages that have been generated up to the current time.
Start Time	Indicates the date and time when the report began to generate.
End Time	Indicates the date and time when the report finished generating.

- You can use any of these fields to search for *spoolheader* records. Place your cursor in the field on which you would like to base your search, and then click **Find** or press **F8**.

A QBE list opens, and a record frame opens to display the item details.

The screenshot shows the Peregrine software interface. At the top, the title bar reads "Peregrine". Below it, the menu bar shows "Main Menu: falcon" and "spool". The toolbar includes icons for "List", a search icon, a print icon, and a dropdown menu. The main area displays a table with the following data:

Number	Page
887	1

Below the table is a record frame with the following details:

OK Cancel Add Save Delete

Name: resolution code listing Number: 887 Page: 1
 Date: 02/12/2004 15:19:53 Printer: sysprint Operator: falcon

02/12/2004 15:
 resolution code listing
 Page: 1
 Operator: falcon Selection: true
 Sequence: Summary
 resolution code resolution
 =====

Figure 1-16: Searching spoolheader records

- 3 You have the following options when viewing the spoolheader records in the QBE list or records frame by clicking the **Options** menu.
 - Mass Add, Mass Update, Mass Unload, Count, Print List, Refresh, Modify Columns, Export to Excel, Export to Text File, Save as Inbox, Breakdown
 - Print, Validity Lookup, Export/Unload, IR Query, Expand Array
- 4 Click **Refresh** to refresh your view of the report.
- 5 Press **F4** to delete the report.
- 6 Click **Back** or press **F3** to finish viewing the report.

How to View Report Schedule Records

Whenever a report is scheduled for background processing, it is first added to the schedule file with a scheduled class of *report*. The schedule definition remains in effect until the report is processed by the report background processor, or until the definition is deleted.

When the background processor begins processing the report, the **class** field is changed from *report* to *blank* and the **status** field is updated to *application running*. When both of these conditions exist, the report is actively being processed by the *report* background processor.

If a repeat interval was specified in the schedule record, the same schedule record is used to re-define when the report should be spooled again. Report Exerciser (the application called by the background processor) takes the original expiration time and adds the repeat interval to it, updates the status to re-scheduled, and changes the application name back to its original setting. See the Report Run Exerciser, Figure 1-7 on page 29.

How to Copy/Rename Reports

To copy and rename reports:

- 1 Access a report, following the procedures discussed in *How to Access Report Writer* on page 8.

In this example, the *ocml.category.print* report opens.
- 2 Click **Edit** or press **F2** to access the edit functions of Report Writer.
- 3 Select **Options > Copy/Rename**.

The Report Entry (*report.prompt*) form opens with the current name of the report in the **Name** field (Figure 1-17).

The screenshot shows a window titled "Report Entry". Inside the window, there are two labels: "Name:" and "Title:". The "Name:" label is followed by a text input field containing the text "ocml.category.print". The "Title:" label is followed by a list box containing the text "Print Line Item Categor". Below the list box are two empty input fields.

Figure 1-17: Using Report Entry to copy and rename reports

- 4 Enter the *new* name of the report over the existing name in the **Name** field. This must be a unique name.
- 5 Enter the *new* report title in the **Title** field.
- 6 Continue with one of the following options.
 - Click **Copy** to create a second copy of the report using a different name. This copies all the Report Writer definitions from the current record to the new report name. If any changes were made to the report title, the edited version is copied to the new report. The existing Report Writer definition will not be changed during the copy process. The formats defined in the original report will not be copied. However, formats can be shared between reports.
 - Click **Rename** to rename the current report. This renames the report, but only the report name and report title are changed. All other definitions for the report remain unchanged.
 - Click **Cancel** or press F3 to cancel the procedure and return to the *report.maint* form. Since the procedure has been canceled, the original report remains unchanged.

How to Delete Reports

To delete reports:

- 1 Access a report, following the procedures discussed in *How to Access Report Writer* on page 8.

In this example, the *ocml.category.print* report opens.

- 2 Click **Delete** or press **F4** to delete the report.

The Report Deletion (*report.prompt.delete*) form opens (Figure 1-18 on page 72).

- 3 Continue with one of the following Delete options.
 - Click **Abort Del** to end the delete process and exit to the *report* form.
 - Click **Rpt Record** to delete only the *report* record.
 - Click **Rpt & Fmts** to delete the *report* record and its *report* formats.
 - Click **All** to delete the *report* record, *report* formats, and all associated application panels. This option is only available for *manual reports*.

Note: When you delete report formats, be sure that the formats defined in this report are not required in any other report that is used.

The screenshot shows a web application window titled "Peregrine". The main menu is "falcon" and the current page is "select delete option". There are four radio buttons at the top: "Rpt Record", "Rpt Fmts", "Abort Del", and "All". The "Rpt Record" option is selected. Below the radio buttons, the "Report Name" is "ocml.category.print" and the "Report Title" is "Print Line Item Category Definition". The section "CHOOSE ONE OF THE FOLLOWING DELETE OPTIONS" contains the following options:

Delete Option	Description
rpt record	Delete ONLY the Report Record
rpt & fmts	Delete the Report Record and its Report Formats
Abort Delete	End the delete process
all	If the Report has associated application panels, delete them, the Report Record and its Report Formats. <i>This option is available only on "manual" reports</i>

Figure 1-18: Report Deletion form with Delete options

How to Use Stacked Queries for Sorting

Report Writer bases report generation on the records retrieved from the primary file by the definition of the *primary query* or the definition of the *stacked queries*. Different rules apply to each of these situations and each has advantages and disadvantages over the other.

If one *primary query* with no *stacked queries* is defined to retrieve records, then the following rules are in effect.

- A work file is *not* created.
- For optimum performance, the *primary query* must be fully keyed.
- If sort fields are defined, then that key (and that key *only*) is used to retrieve records from the database.

Note: If the *primary query* is *not* fully keyed based on the sort key, turnaround time will be degraded.

For example, the following keys are defined for a file:

```
category
status
open.time
assignment
category
```

and the sort fields for a report are:

```
open.time
assignment
category
```

then the query:

```
category="systems"
and
status~="closed" (Unix)
```

This is *not* fully keyed, because the sort fields will force the system to use the second key to retrieve the records. Therefore, the query that is defined is partially keyed and each record retrieved from the database (in this case all records appearing in the key list for the second key) have to be expanded to determine if the data record meets the search requirement.

Primary Query Option

Advantage of Using a Primary Query Over Stacked Queries

Choosing to use the Primary Query option over the Stacked Queries option is advantageous under the following circumstances.

- A query of *true* is defined and either:
 - No sort fields are defined (in this case the first key in the key list would be used to retrieve the data records).
- or-
- The sort fields are defined as a key.
- The primary query and the sort fields reference the same key and the primary query is fully keyed.
- The primary query is fully keyed and no sort fields are defined.

Stacked Queries Option

Advantage of Using Stacked Queries

Choosing to use the Stacked Queries option is advantageous under the following circumstances.

- The sorting requirements for a report are not defined as a key in the primary file's dbdict.
- The primary query does not reference the same key as the sort fields.
- The reporting requirements are such that one fully keyed query cannot be defined.

Example of Stacked Queries Option

An example of how to use the Stacked Queries option is as follows.

Example: The requirements of this report are to list all open Incident tickets for the categories of **lines**, **modems**, and **enduser**. The sort requirements are by **category**, **assignment**, **status**, and **open.time**. The primary file is **probsummary** and the following keys are defined:

```

number
*****

flag
category
subcategory
company
*****

logical.name
*****

```

The following specifications would be in effect for the report if the Stacked Queries option was used.

Stacked Queries Specifications

Unique sort sequence.

primary file: probsummary

primary query: true

stacked queries: category="lines" and
status="closed"

category="modems" and
status="closed"

category="enduser" and
status="closed"

Note that each of the queries is fully keyed. While this is not a requirement, it is strongly recommended that the queries be defined for stacked queries in a way that allows the system to meet all or most of the query requirements from a key list rather than having to expand records retrieved from a particular key to satisfy the query requirements.

Sort Fields

category
assignment
status
open.time

Note: These sort fields are *not* defined as a key in the primary file, but they will be defined as a key in the new temporary file.

If the report was run at approximately 15:00 hours on 02/07/02, the name of the work file would be created with a date stamp:

workprobsummary0207021500

Rules to Follow When Defining Stacked Queries

If one or more stacked queries are defined for a report, the following rules are in effect.

Note: Stacked queries are separate and distinct from secondary queries.

- A work file *is* created.
- Any Format Control definitions specified for any formats used in the report are still in effect.
- Any subtotalling requirements defined for the report are still in effect.
- The sort fields do not have to be defined as a key in the primary file's dbdict, but *must* be defined as fields.
- If structure names are used in one sort field, structure names must be used in all sort fields.
- The following validations are performed on all sort fields (any errors are logged in the msglog file):
 - The structure name (if any) referenced in each sort field must be valid.
 - The sort field must belong to the indicated structure (if one is specified).
 - The sort field must be a **scalar number, character, logical, or date/time** field. *No other sort field types are allowed.* If other types of fields are specified, the sort will be aborted.
- Each stacked query is executed against the primary file and all records are temporarily stored in the sortwork file.

- The primary query is executed against the **sortwork** file and under normal circumstances is defined as *true*.
- The **work** file that is created is a logical file defined with the following naming standard:
workxxxmddyhhmss

Naming Standard Definitions

work =	Constant
xxx =	The name of the primary file
mmddyhhmss =	The date and time that the work file was created.

- The **work** file is defined in physical file FILE3. Therefore, caution should be exercised when using stacked queries which will cause all records from any large file to be read and placed in the **sortwork** file.
- The **work** file will be scheduled for deletion two hours after its creation with a class of **report** and an operator name of **sortsubr**. It is recommended that the system administrator periodically review the dbdict file to ensure that all work files have been deleted. Be sure to only delete those work files that completely follow the naming standards of the sort work files.
- The sort routine copies all unique keys from the **primary** file to the **sortwork** file and adds the sort fields as **nulls&duplicates** key. All **no nulls** and **no duplicates** keys that display in the key list before the first unique key are copied to the **sortwork** file. If **no unique**, **no nulls**, or **no duplicates** keys exist, the first key in the key list is copied to the **sortwork** file.
- If the same record is retrieved using different queries and at least one unique key is defined for the **sortwork** file, the duplicate record is not added to the **sortwork** file.
- All error messages issued from the sort routine are logged in the **msglog** file.
- A message stating how many records were added to the **sortwork** file, based on the execution of each stacked query, is issued to the **msglog** file.

Report Writer Options Menu Selections

The following options are available within the various Report Writer forms.

Option	Description
Add	Allows you to modify selected records in a file and add them to the database, while leaving the original records in place.
Copy/Rename	Allows you to create a new record by copying an existing record and renaming the new record.
Delete	Allows you to delete all the records appearing in a record (or QBE) list. Query for a certain subset of records and select mass delete. You are prompted to confirm the action.
Export to Excel	Exports all the records displayed in the record or QBE list directly to Microsoft Excel. Basically, you can export information to any product that has DDE (Dynamic Data Exchange) support. This DDE function requires Excel 95 or later. Note: This option is available only on Windows clients.
Export to Text File	Exports all the records displayed in the record or QBE list to an external file in text format. Choose a file name and text delimiter in the dialog box displayed. The file is exported to the RUN directory in your client installation. This option is not part of the DDE support and can be run from clients other than those operating in Windows.
Forms Designer	Allows you to go from the <i>report</i> form into the Forms Designer utility to change the format of a new or existing report.
Maintenance Schedule	Allows you to modify the current schedule for this report.
Mass Unload	Allows you to unload all the records displayed in a record or QBE list to an external file.
Modify Columns	Allows you to edit column headings in a record or QBE list.
Print	Allows you to print either the record/QBE list or all the records.
Save As Inbox	Allows you to save the current list as an inbox.
Set Query	Brings you to the <i>report.query.maint</i> form to add a query or modify an existing query.

Option	Description
Totals	Brings you to the <i>report.totals</i> form of the existing report.
Update	Allows you to update the current record.
View Spooled Reports	Allows you to view spooled reports. When you select to View Spooled Reports , the <i>spoolheader</i> form opens. From here, you click Find to search for records.

