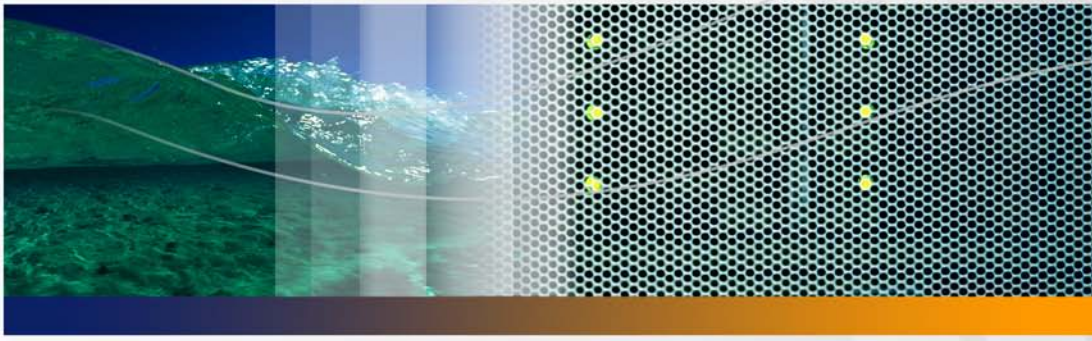
Peregrine Systems, Inc.

# ServiceCenter® 6.1

Product Documentation



# Report Writer Guide

Part No. DSC-61-EN54

Peregrine SYSTEMS

© Copyright 2005 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, 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. This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim 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. This product includes software developed by the OpenSSL Project for use in the OpenSSL toolkit. (http://www.openssl.org/). OpenSSL software copyright 1998-2005 The OpenSSL Project. All rights reserved. This product includes software developed by the MX4J project (http://mx4j.sourceforge.net). 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 © Copyright 2000, 2004 IBM Corporation and others. All rights reserved. Copyright 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/>). Portions copyright © Daniel G. Hyams, 1998. cbg.editor Eclipse plugin copyright © 2002, Chris Grindstaff. Part of the software embedded in this product is gSOAP software. Portions created by gSOAP are copyright © 2001-2004 Robert A. van Engelen, Genivia inc. All Rights Reserved. This product leverages open source components. Please contact Peregrine Systems (support@peregrine.com) to be directed to the appropriate site to access the unmodified open source code, or to obtain the modified open source code, as applicable, for the relevant open source components.

ReportCenter includes software developed by the Apache Software Foundation (http://www.apache.org). Copyright © 2000, The Apache Software Foundation. All rights reserved.

This product includes software developed by the Apache Software Foundation (http://www.apache.org/). This product also contains software developed by: Sun Microsystems, Inc., Netscape Communications Corporation, and InstallShield Software Corporation.

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 6.1 of the licensed program.

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

PEREGRINE

# Contents


Introducing Report Writer . . . . . 7

Knowledge requirements . . . . . 7

Need further assistance? . . . . . 8

Customer Support . . . . . 8

Documentation Web site . . . . . 8

Education Services Web Site . . . . . 9

Chapter 1 Report Writer Basics . . . . . 11

Overview . . . . . 11

How do I access a report?. . . . . 12

How do I copy or rename reports? . . . . . 14

How do I delete reports? . . . . . 15

How do I create a new report? . . . . . 16

Chapter 2 Using Report Forms . . . . . 19

How do I create or edit report forms? . . . . . 20

Creating Query forms for reports . . . . . 21

How do I create Query forms? . . . . . 21

Using initial forms with reports . . . . . 27
How do I access a form in Forms Designer? . . . . . 27
How do I use an initial form? . . . . . 27
Using page breaks and report totals . . . . . 28
How do I access the Totals section of a report? . . . . . 28
Sample Report 1: Using Simple subtotals . . . . . 31
Sample Report 2: Using subtotals, summary totals, and grand totals . 33

Chapter 3 Using Format Control in Reports . . . . . 37
Accessing Format Control . . . . . 37
How do I access Format Control? . . . . . 38
Format Control options . . . . . 38
Sample Report 3: Using Format Control for Calculations . . . . . 39
Sample report 4: Using Format Control for subheaders . . . . . 42
Sample Report 5: Using Format Control in a Matrix report . . . . . 46

Chapter 4 Using Stacked Queries . . . . . 53
When should I choose the Primary Query option? . . . . . 53
Rules for primary queries . . . . . 54
When should I choose the Stacked Queries option? . . . . . 55
Rules for stacked queries . . . . . 55
Stacked Queries Example . . . . . 56

Chapter 5 Running Reports . . . . . 59
Running reports in foreground . . . . . 59
How do I run report in the foreground? . . . . . 59

Background Processing . . . . . 60
How do I run a report in the background? . . . . . 61
Viewing and editing report schedules . . . . . 64
How do I edit report schedules? . . . . . 64
Viewing reports . . . . . 65
How do I view a report online? . . . . . 65
Index . . . . . 67

PEREGRINE

# Introducing Report Writer

Welcome to the ServiceCenter Report Writer guide. Report Writer is a component of the ServiceCenter toolkit. This guide tells you how to run existing reports and create new reports using Report Writer.

This guide contains information on:


- Report Writer Basics on page 11
- Using Report Forms on page 19
- Using Format Control in Reports on page 37
- Using Stacked Queries on page 53
- Running Reports on page 59


## Knowledge requirements

To run Report Writer reports, you need a working knowledge of your ServiceCenter system. To create new Report Writer reports you need a working knowledge of your DBMS, ServiceCenter applications and utilities.

- For information about a particular RDBMS, refer to the documentation for your database.
- For database configuration information, refer to the ServiceCenter Database Conversion and RDBMS Support guide.
- For information about P4, ServiceCenter administration and configuration, and customizing ServiceCenter, refer to the ServiceCenter online help.

# 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 the Peregrine CenterPoint Web site.

### To contact customer support

1. In a browser, navigate to *http://support.peregrine.com*.
2. Log in with your user name and password.
3. Follow the directions on the site to find your answer. The first place to search is the KnowledgeBase, which contains informational articles about all categories of Peregrine products.
4. 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.

### To view the document listing

1. In a browser, navigate to *http://support.peregrine.com*.
2. Log in with your login user name and password.
3. Select either **Documentation** or **Release Notes** at the top of the page.
4. Open the **ServiceCenter** link.

5 Open a product version link to display a list of documents that are available for that version of ServiceCenter.

6 Documents may be available in multiple languages. Click **Download** to download the PDF file in the language you prefer.

You can view PDF files using Acrobat Reader, which is available on the Customer Support Web site and through Adobe at *http://www.adobe.com*.

# Education Services Web Site

Peregrine Systems offers classroom training anywhere in the world, as well as "at-your-desk" training using the Internet. For a complete listing of Peregrine's training courses, refer to the following web site:

*http://www.peregrine.com/education*

You can also call Peregrine Education Services at +1 858.794.5009.

CHAPTER 1

# Report Writer Basics

Report Writer is a component of the ServiceCenter toolkit. You can use it to query for existing reports, and to create simple or complex reports that you can modify at any time.

## Overview

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

You can base a report on one or more database tables. If you use several tables, you will need to specify a primary file. You can define the data selection and totaling criteria as well as the forms used to display the report.

# How do I access a report?

### To access an existing a report using Report Writer

1 Open Report Writer.

- Open **ToolKit** > **Report Writer.**
- Type `rw` on the ServiceCenter command line and press **Enter**.

The Report Writer Entry form (report.prompt.rw) opens.

2 Select one of the following options.

- Click **Search** or press **Enter** without specifying any values, to pull up a list of existing report records.
- Type an existing report name or primary file name and click **Search** or press **Enter**, to find reports with that match your search criteria.

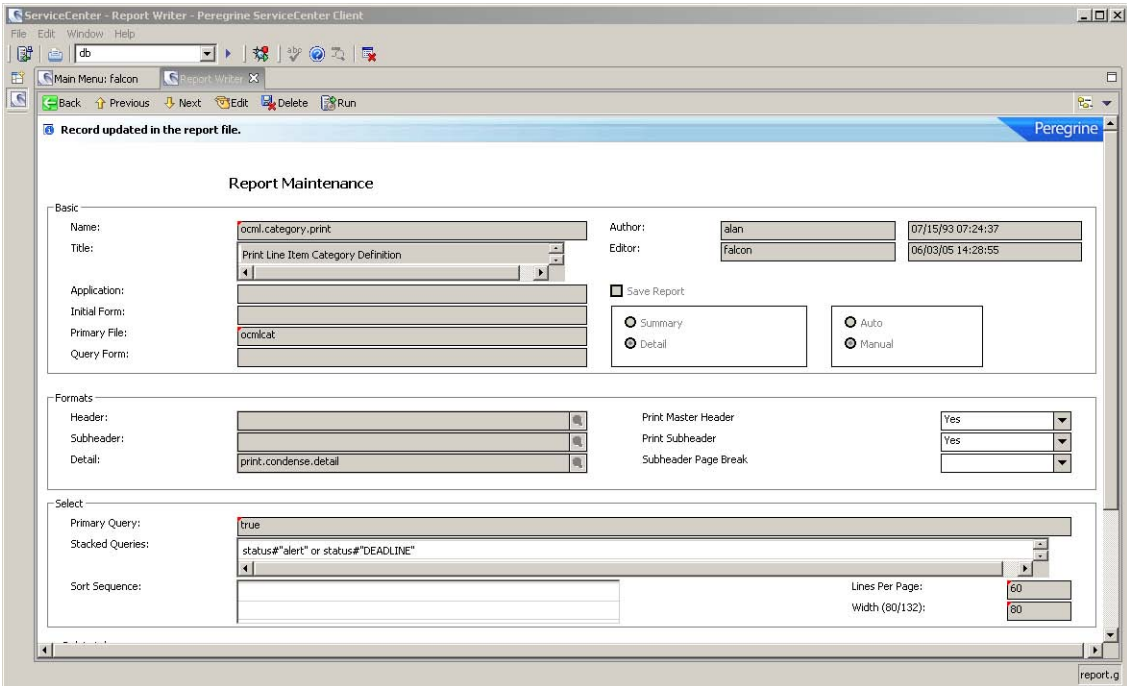
  If the specified query produces only one record, that report record opens on the Report Maintenance form.

  If the specified query selects more than one record, Report Writer displays a list of report records. Scroll through the list. Place your cursor on any record and double-click to select the record.

3 Click **Back** to exit Report Writer.

## The Report Maintenance form

When you open an existing report, Report Writer displays it in the Report Maintenance form (*report)*.




The fields in this form are described in the ServiceCenter field help.

- To view the help for a field, put your cursor in the field and press **Ctrl+H** or select **Help** > **Help on Field**.

The available options may vary depending on the preferences you have selected, the record you select, and your permissions.

| Button | Option | Description |
|---|---|---|
| | Update | Saves changes to the current record. |
| | Back | Returns to the previously displayed form. |
| | Delete | Deletes the currently displayed query record. |

| Button | Option | Description |
|---|---|---|
| | Run | Opens the Run Report Exerciser, enabling you to run a report in the foreground or background. It saves new reports and modifications made to existing reports. For more information, see Running Reports on page 59. |
| | Previous | Displays the previous report on the list. |
| | Next | Displays the next report on the list. |
| | Edit | Opens the Report Maintenance Editing form, enabling you to edit the report. |
| | View Spooled Reports | Opens the spoolheader form, enabling you to view spooled reports. For more information, see Viewing reports on page 65. |
| | Forms Designer | Opens Forms Designer, enabling you to edit the form for the report. For more information, refer to the ServiceCenter online help. |
| | Create RAD | Opens the RAD Editor, enabling you to create or edit RAD code for the report. |
| | Maintenance Schedule | Open the form, enabling you to modify the current schedule for this report. For more information, see Viewing and editing report schedules on page 64. |
| | Close Application | Closes the current application. |

# How do I copy or rename reports?

### To copy or rename reports

1. Access the report.
   For this example, open the *ocml.category.print* report.

2. Click **Edit** to access the edit functions of Report Writer.

3. Open **Options** > **Copy/Rename**.
   The Report Writer Entry form opens with the current name of the report in the **Name** field.

4. Type new report's name over the existing name in the **Name** field.
   This must be a unique name.

5 Type the new report's title in the **Title** field.

6 Continue with one of the following options.

| Button | Option | Result |
|---|---|---|
| | Copy | Creates a copy of the report using a different name.<br>This action copies all definitions from the current record to the new report. If you make any changes to the fields, Report Writer copies the edited version to the new report. Report Writer does not change the existing definition during the copy process. Although reports can share forms, Report Writer does not copy the forms defined in the original report. |
| | Rename | Renames the current report.<br>This action changes only the report name and report title. All other definitions for the report remain unchanged. |
| | Cancel | Cancels the procedure and return to the Report Maintenance form.<br>This cancels all actions. The original report remains unchanged. |

# How do I delete reports?

### To delete reports

1 Access the report.
For this example, open the *ocml.category.print* report.

2 Click **Delete** to delete the report.
The Report Deletion (*report.prompt.delete*) form opens.

3 Continue with one of the following Delete options.

- Select **Abort Del** to end the delete process and exit to the report form.
- Select **Rpt Record** to delete only the report record.
- Select **Rpt Fmt** to delete the report record and its report forms.
- Select **All** to delete the report record, report forms, and all associated application panels. This option is only available for manual reports.

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

# How do I create a new report?

### To create a new report

1 Open **ToolKit** > **Report Writer**.
- or -
Type `rw` on a ServiceCenter command line and press **Enter**.
The Report Writer Entry form opens.

2 Type a unique name for the new report in the **Report** field. Do not use special characters other than "." or blank spaces.

For this example name the report, `problem.category.report`.

**Note:** Use the text before the first "." to indicate the primary file for the report. If there are no ".", ServiceCenter does not associate a primary file with the report.

For example, if you name the report problem.category.report, ServiceCenter assumes that *problem* is the name of the primary file. If you type `problemcategory` without a separating ".", Report Writer does not make that assumption.

3 Type the name of the primary file for the report.

4 Click **Create New Report**.

ServiceCenter checks to see if the report form already exists.

- If the report form already exists, Report Writer displays it in the Report Maintenance form to enable you to make changes, or copy data to create a new report.
- If the report form does not exist, Report Writer displays the new report in the Report Maintenance form and populates the **Name**, **Primary File**, **Header**, and **Details** fields based on the data you entered.

5 Fill in the fields and click **OK**. For more information, see The Report Maintenance form on page 13 and The Report Maintenance Editing form on page 17.

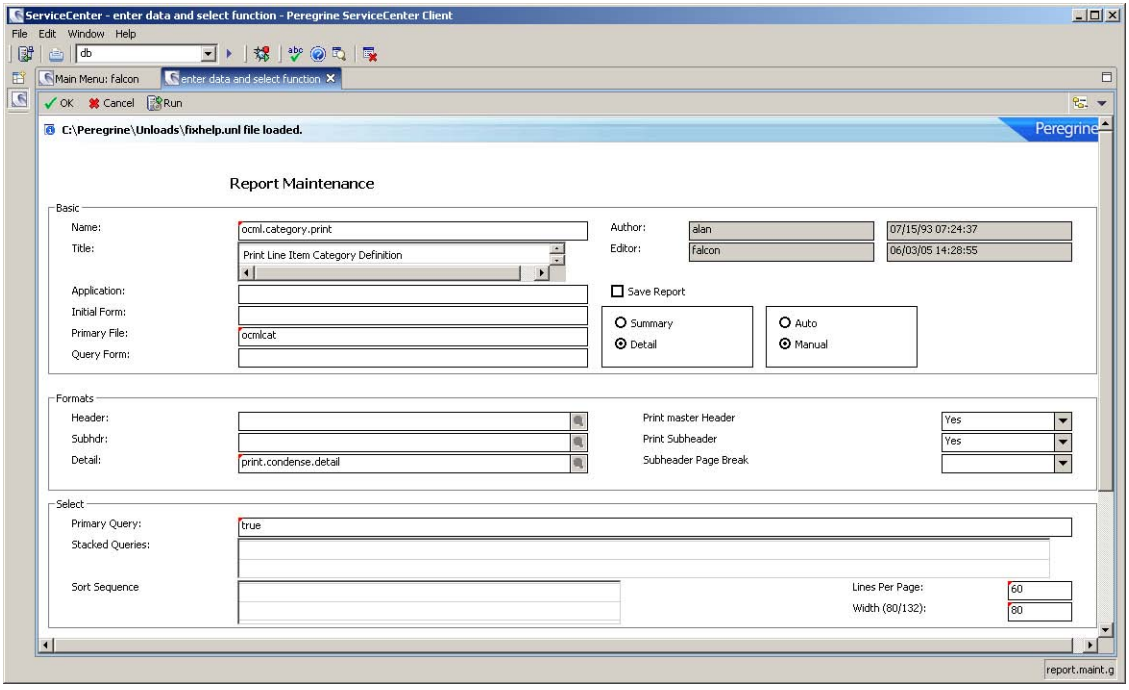
**Note:** To set the width of a report, complete the **Page Width** field of the **Client Printing** structure in the user's operator record. For more information, refer to the ServiceCenter online help.

The maximum width of a report is 132 characters. If the forms defined for a report exceed this limit, Report Writer generates them with the specified width. However, when passing the report to the printer, it truncates them 132 characters. You can change this setting in the *config* record for the printer used to print the report.

6 Create the header and detail forms for the report, if they do not already exist. For more information, see How do I create or edit report forms? on page 20.

## The Report Maintenance Editing form

When you create a new report, Report Writer displays Report Writer displays it in the Report Maintenance form (report.maint).



The fields in this form are described in the ServiceCenter field help.

- To view the help for a field, put your cursor in the field and press **Ctrl+H** or select **Help** > **Help on Field**.

The available options may vary depending on the preferences you have selected, the record you select, and your permissions.

| Button | Option | Description |
|---|---|---|
| | OK | Saves the report and returns you to the previous screen. |
| | Cancel | Cancels any changes and returns you to the previous screen. |
| | Run | Opens the Run Report Exerciser, enabling you to run a report in the foreground or background. It saves new reports and modifications made to existing reports. For more information, see Running Reports on page 59. |
| | Forms Designer | Opens Forms Designer, enabling you to edit the form of a report. For more information, refer to the ServiceCenter online help. |
| | Set Query | Opens the Report Query Maintenance screen, enabling you to edit or define the primary query for the selected report. For more information, see Creating Query forms for reports on page 21. |
| | Totals | Opens the Report Totals screen, to enable you to define totals parameters. For more information, see Using initial forms with reports on page 27. |
| | Copy/Rename | Enables you to create a new record by copying an existing record and giving it a new name. For more information, see How do I copy or rename reports? on page 14. |

# CHAPTER 2 Using Report Forms

When you create a report, Report Writer uses several forms to generate it.

This table describes the report forms.

| Report Form | Description |
|---|---|
| Initial | Collects data from the user at runtime. |
| Header | Displays the report titles or headings.<br>Report Writer prints this form at the top of every page. It usually contains column headings that explain the fields in the detail section of the report. |
| Detail | Displays the values of database fields.<br>The detail form comprises the input fields, positioned in a columnar arrangement immediately beneath the corresponding headings.<br>The names of the input fields must reflect the exact names used in the respective database tables or variables that are defined through Format Control or Totals Parameters definitions.<br>You can use the same detail form for multiple reports. |
| Subheader | Displays information immediately preceding detail information extracted from a secondary file.<br>To use subheader reports, you must define a Format Control record for the detail form, to identify the secondary file to use to extract information for the detail form. |
| Totals | Display summary totals of the data gathered. |

# How do I create or edit report forms?

### To create or edit a report form

1 Open an existing report following the instructions in How do I access a report? on page 12.

For this example, open `problem.category`. This is the report created in How do I create a new report? on page 16.)

2 Click **Edit** to open then editing screen and add any necessary information. The Report Maintenance form opens.

3 Edit the fields, based on the following table and the information in The Report Maintenance form on page 13.

| Field | Description |
|---|---|
| Detail | problsummary.category.detail |
| Sort Sequence | assignment<br>This example report does not use stacked queries, therefore the sort sequence field must be a key in the primary filename (probsummary) dbdict. |
| Width (80/132) | 80 |

4 When filling in a detail form for a report, you must define the **Input** fields. You may also define **Caption** fields.

5 Open **Options** > **Forms Designer**.

If the **Name** field is blank, Forms Designer opens a prompt where you can name your report.

a Type the detail form name at the **Name** field.

For this example, type `probsummary.category.detail`.

b Associate the form with the primary file of the report.

For this example, use `probsummary`.

6 Fill in the detail form, following normal Forms Designer procedures. For more information, refer to the ServiceCenter online help.

7 Click **Add**.

**Note:** Edit the field names to reflect their dbdict field names. To ensure that the input field names are correct, you can associate the form with the `primary` file of the report. As you define each input field, check the field name to verify that the field does exist in the `primary` file.

8 Click **Cancel** to close the edit field window.

9 Click **OK** or **Back** until you return to the Report Maintenance form.

10 Save the newly-created form.

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

11 To run this report, follow the instruction in Running Reports on page 59.

# Creating Query forms for reports

Report Writer provides the option to define a set of queries for a report, which you can then use as the Primary Query when initially scheduling the report to run. ServiceCenter stores these queries by report name and user ID. You can access them for viewing and updating. These stored queries do not over write the original Primary Query defined in the report record.

## How do I create Query forms?

**To create a query form for a report and establish stored queries**

1 Open an existing report following the instructions in How do I access a report? on page 12.

For this example, open `problem.category.report.` This is the report created in How do I create a new report? on page 16. The Report form opens.

2 Click **Edit** to access the edit functions of Report Writer.

3 Specify a Q**uery Format** name.
If you leave the **Query Format** field blank, Report Writer takes the value in the **Primary File** field as the form to use when creating queries.

For this example leave the this field blank.

4 Open **Options** > **Set Query** to access the query form and begin the process of setting queries for the report. The Report Query Maintenance form opens. For a description of this form, see The Report Query Maintenance form on page 25.

- If report queries already exist for this report, a record list of those queries opens. From this list, you can select a query for maintenance.
- If there no report queries exist for this report, ServiceCenter prompts you to add a query.

5 Select a record from the list and click Edit to make changes.

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

a Type the name of the query in the **Name** field and the query into the **Query** field.

For this example, type `Alert Status` in the **Name** field.

b Type the query you want to use as the primary query in the **Query** field.

For this example, type the following in the **Query** field.

```
status#"alert" or status#"DEADLINE"
```

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

7 To begin the process of entering a report query, click **Build**.

The associated form opens. For this example, the *probsummary* form opens.

Report Writer displays the form and the search arguments you specify in the **Query** field of the Report Query Maintenance form. For more information, see Field descriptions for the probsummary form on page 26.

8 Type data that will satisfy your search. The more fields you fill in, the narrower your search will be.

For this example, type the following in the **Category** field.
`=network`

9 Click **Query Window** to open the query form. You can view and edit the SQL translation of this query in this form.
- or -
Press **Enter** to translate the query into SQL. ServiceCenter redisplays the form with that SQL expression in the **Query** field.

For this example, click **Query Window**.

**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.

You must make the necessary modification to the translated SQL to accommodate these or any other special query needs.

If the query is correct, press **Enter**.
- or -
If the query shown in the Report Query Maintenance form is not the desired query, make changes as needed and click **Update**.

For this example, the desired query, which will select all records from the `probsummary` file that have a *category* value of *network*, is displayed. Therefore, press Enter.

ServiceCenter displays the translated SQL expression in the **Query** field.

10 Specify a unique name for the query.

For this example, type the following in the **Name** field.

```
network
```

11 Optionally, you can modify the SQL expression shown in the **Query** field.

For this example, add the following to the expression in the **Query** field.

```
and flag=true
```

12 Click **Add** to add a new query record, or click **Update** to update an existing record.

13 Click **Back** to get back to the to the Report Maintenance form.
- or -
Click **Run** to run the new report query. For more information, see Running Reports on page 59.

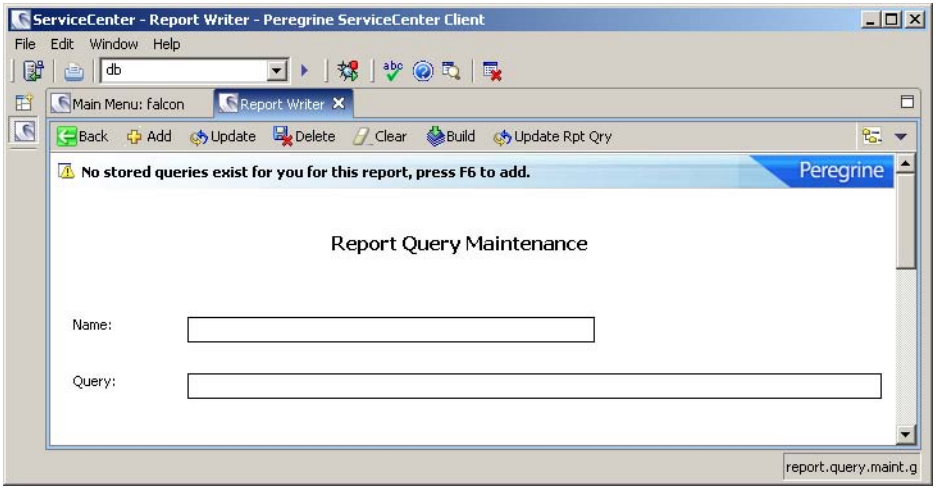
A record list of one or more queries opens.

- Double-click the query name of one of the query records listed.
- The Run Report Exerciser form opens, enabling you to schedule the report to run in either the background or foreground.

  While in the Run Report Exerciser form, the query you defined in the previous steps is now in the **Primary File Query** field. However, the original Primary Query defined in the report record itself is not affected.

## The Report Query Maintenance form

When you add queries to a report, Report Writer displays your options in the Report Query Maintenance form (report.query.maint).




The fields in this form are described in the ServiceCenter field help.

- To view the help for a field, put your cursor in the field and press **Ctrl+H** or select **Help** > **Help on Field**.

The available options may vary depending on the preferences you have selected, the record you select, and your permissions

| Button | Option | Description |
|---|---|---|
| | Add | Enables you to modify a selected record and add it to the database, while leaving the original record in place. |
| | Update | Saves changes to the current record. |
| | Back | Returns to the previously displayed form. |
| | Delete | Deletes the currently displayed query record. |
| | Build | Displays the form defined in the Query Form. If you do not specify a form, ServiceCenter displays the form with the same name as the primary file name of the report (if any.) |

| Button | Option | Description |
|---|---|---|
| | Clear | Clears all displayed fields. |
| | Update Report Query | Copies the currently displayed query and overlays the Primary Query of the report you are editing. |

## Field descriptions for the probsummary form

The field descriptions in the *probsummary* form describe how you to set up your data to satisfy your search. The Header Information structure delineates how to print each record within the query.

| Field | Description |
|---|---|
| *Header Information structure* | |
| Incident No | Use this field to query for a specified Incident number. Leave this field blank to return records with any incident number. |
| Status | Use this field to query for a status. Leave this field blank to return records with any status. |
| Category | Use this field to query for incidents classified by this the category. Leave this field blank to return records with any category. |
| Assigned To | Use this field to query for incidents assigned to this person. Leave this field blank to return records assigned to any person. |
| Priority | Use this field to query for incidents with the Priority (1 through 4). Leave this field blank to return records with any priority. |
| Change No. | Use this field to query for incidents with this the unique change record number. Leave this field blank to return records with any change number. |
| Page | Use this field to query for a specific page number of the printed report. Leave this field blank to return all pages. |
| Reassign Count | Use this field to query for incidents tickets that have been reassigned the specified number of times. Leave this field blank to return records that have been reassigned any number of times. |
| Opened | Use this field to query for incidents opened on this date. Leave this field blank to return records opened on any date. |
| Alert | Use this field to query for the alert status that was generated, if an alert has been generated for this record. Leave this field blank to return records with any alert status. |
| Updated | Use this field to query for the date the record was last updated. Leave this field blank to return records updated on any date. |
| Closed | Use this field to query for the date the record was closed. Leave this field blank to return records closed any date |

| Field | Description |
|---|---|
| Logical Name | Use this field to query for the logical file name of the device associated with the record. Leave this field blank to return records with any logical name. |
| Location | Use this field to query for a particular company location. Leave this field blank to return records with any location. |
| Contact Name | Use this field to query for a particular contact name. Leave this field blank to return records with any contact name. |
| Service Contact | Use this field to query for a particular service contact. Leave this field blank to return records with any service contact. |

# Using initial forms with reports

An initial form enables you to enter data for Report Writer to use to develop the query for the report. If you want to retrieve data from the user at runtime, you must create an Initial form.

For example, if the report requires a date and time range, you can use an initial form to enable the user to enter this range rather than entering a complete SQL query in the **Primary Query** field.

## How do I access a form in Forms Designer?

### To access a form in Forms Designer

- From the Report Maintenance form or Report Totals form, open **Options** > **Forms Designer**.

- From the main menu or system navigator, open **Toolkit** > **Forms Designer**.

## How do I use an initial form?

### To specify an initial form for a report

1 Access the form in forms designer.

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

2 Type the form name in the **Initial Format** field.

3 Create the form, using the usual Forms Designer procedures. For more information, refer to the ServiceCenter online help.

**Note:** The **Input** field names for the fields displayed when creating your new form in Forms Designer are `$start` and `$end`.

4 To include information from the initial form on the report, change the Primary Query statement to the following.

```
number> $start and number <$end
```

5 Click **Run** to run the report. For more information, see Running Reports on page 59.

**Note:** You cannot run reports that use the initial form option in the background. Therefore, the **Run Report in Background** field is defaulted to f (false).

6 Click **Run** to complete the request.

# Using 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. You can print these statistics according to user - defined forms at various phases of report generation based on user - defined control/condition breaks. You can use variations of Totals Parameters with or without Format Control definitions. Format Control definitions enable you to create a wide variety of report statistics.

## How do I access the Totals section of a report?

This section demonstrates accessing the Totals section of a report and defines the various Totals Parameters, including referencing them and creating the forms used to print accumulated statistics.

### To access the Totals section of a report

1 Open **ToolKit** > **Report Writer**
- or -
Type `rw` on a ServiceCenter command line and press **Enter**.
The Report Writer Entry form opens.

2 Type `problem.category` in the **Report** field.
The Report form opens.

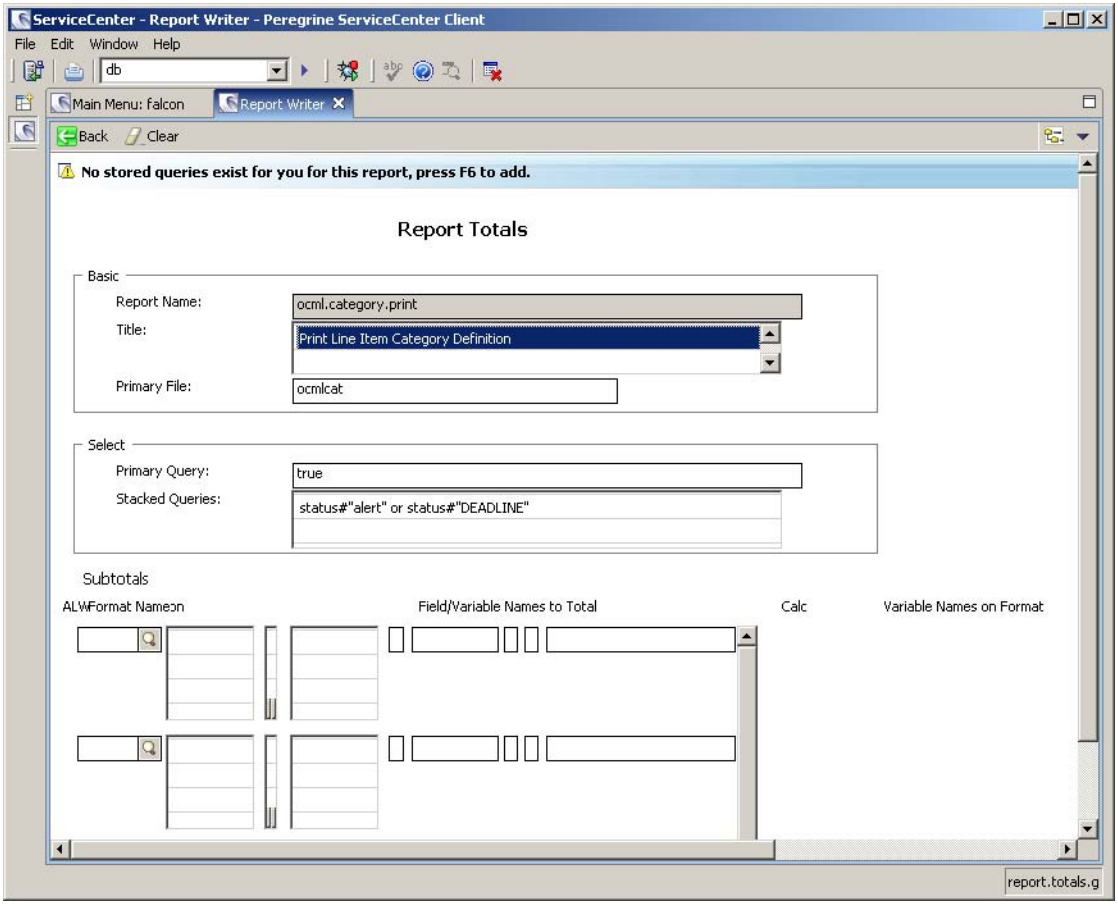
Click **Edit** to access the edit functions of Report Writer.
The Report Maintenance form opens. See the The Report Maintenance Editing form on page 17.

3 Open **Options** > **Totals** to access the Report Totals (report.totals) form. This form is described in The Report Totals form on page 30.

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.

## The Report Totals form

When you create a totals or subtotals report, Report Writer displays it in the Report Totals form (report.totals).




The fields in this form are described in the ServiceCenter field help.

- To view the help for a field, put your cursor in the field and press **Ctrl+H** or select **Help > Help on Field**.

The available options may vary depending on the preferences you have selected, the record you select, and your permissions

| Button | Option | Description |
|---|---|---|
| | Back | Returns to the previously displayed form. |
| | Clear | Clears the data displayed on the form. |
| | Forms Designer | Opens Forms Designer, enabling you to edit the form of a report. For more information, see the ServiceCenter online help. |

# Sample Report 1: Using Simple subtotals

This sample report demonstrates how to define the Totals parameters to print a subtotals form. It is a continuation of the from created in How do I create a new report? on page 16.

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

### To create a subtotals form

1 Access the totals section of the report you want to create a totals form for. For more information, see How do I access the Totals section of a report? on page 28.

For this example access the totals section of the *problem.category* report.

2 Type the name of the form want to use to print the subtotal in the **Form Name** field of the **Totals** structure.

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

Report Writer checks for a form called *probsummary.demo.totals* in the `format` table. If the form already exists, it opens.

If the form does not exist, open **Options** > **Forms Designer** and create the form using normal Forms Designer procedures. For more information, refer to the ServiceCenter online help.

3 Create each field, using the input properties described in this table.

Input properties for the subtotals form:

| Field | Description |
|---|---|
| report.count | Contains the total number of detail records processed by the report and is normally printed on a grand totals form, but may be printed on any totals form to show a running total. |
| report.subcount | 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 subtotals form, 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. |
| report.sumflds | Contains the value of each field that caused a control break. This field is normally used in conjunction with **report.subcount** for summary totals. |
| report.subfld | Contains the value of the field entered in **Reset Field/Variable Name** at the time a control/condition break is detected. |
| $rw.dummy | A variable defined by Report Writer that contains NULL characters. |
| $rw.months | Contains full month names. January is the first element with December being the twelfth element. |
| $rw.mon | Contains abbreviated (three-character) month names. Jan is the first array element with Dec being the twelfth element. |
| $rpt.date | A variable defined by Report Writer that contains the date and time when the report was run. |
| $page | A variable defined by Report Writer that contains the page of the output of the report. |

4 Click **OK** to save your new form before exiting the Forms Designer to save the changes you made to the form.

5 Scroll to the left to enter parameters in other fields on this form.

**Note:** Any time you scroll left or right while in this form, 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, type its name in the **Reset/Field Variable Name** field.

This causes the subtotals form (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 form.

8 Type `true` in the **Always** field.

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

9 Click **End** to return to the Report Maintenance form.

**Note:** If you would like to erase all values in the Report Totals form, click **Clear Fields**.

A confirmation prompt will display, allowing you to either click **Clear Fields** to continue clearing the fields or **Abort** to cancel deleting the totals information.

10 To run this report, follow the instruction in Running Reports on page 59.

# Sample Report 2: Using subtotals, summary totals, and grand totals

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

- Summary totals display the same statistics printed in each subtotal, but in a summarized form. Changing the **Summary/Detail** field on the Report Maintenance form causes only the totals portions (subtotals, summary totals, and grand totals) of the report to print. The detail form does not print.
- Grand totals show the total number of detail records selected during report generation.

### To define the totals parameters

1 Create the report. For this example, use the following parameters.

| Field | Value |
|---|---|
| Report Name | operator.count |
| Report Title | Monthly Inventory |
| Auto/Manual | Auto |
| Summary/Detail | Detail |
| Primary File | operator |
| Header Format Name | operator.count.header |
| Detail Format Name | operator.count.detail |
| Primary Query | true |
| Sequence of Report | model |

For instructions on creating a report, see How do I create a new report? on page 16.

2 ServiceCenter checks for a form the matches the name you specified in the Detail Format Name fields.

- If the form exists, it open.
- If the form does not exist, Forms Designer opens.

3 Create the detail form using following input fields. For more information, refer to the ServiceCenter online help.

model

logical.name

serial.no

location

4 Add captions to identify the fields.

5 When you are finished adding the report, click **Edit Rpt** to access the Edit functions.

6 Click **Totals** to access the Report Totals form.

7 Type the name of the form to use to print the subtotal in the **Subtotals Format Name** field.

For this example, type `operator.count.subtotal`.

**Note:** You can create the subtotals form through normal Forms Designer procedures either before entering the Totals Parameters or during the Totals Parameters definition process.

In this example, we create the subtotals form *operator.count.subtotal* from the Report Totals form.

8 Position the cursor on the **Totals Format Name** field and open **Options** > **Forms Designer**.

ServiceCenter checks for a form called *operator.count.subtotal* in the `format` table.

- If that form already exists, it opens.
- If the form does not exist, a blank screen opens, allowing you to fill in the form using the normal Forms Designer procedures. For more information, refer to the ServiceCenter online help.

9 Edit the fields to update the form. For the example use the following values.

| Caption | Input |
|---|---|
| Subtotals for: | `report.subfield` |
| No. of recs: | report.subcount,1 |
| | `$rw.dummy` |

10 Click **OK** to return to the Report Totals form.

11 To force a page break after each subtotals form prints, type `true` in the **New Page** field.

12 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**.

13 To define the field by which you will be subtotaling, type its name in the **Reset/Field Variable Name** field. This causes the subtotals form (*operator.count.subtot*) to print every time the value of the **Model** field changes.

   For this example, type `model`.

14 Scroll to the left again to enter parameters in the third section of this form

15 To run this report, follow the instruction in Running Reports on page 59.

# CHAPTER 3 Using Format Control in Reports

Report Writer uses several forms. Each of these forms can have Format Control records for Report Writer to run when executing your report.

This table describes format control on the report forms and how to use it.

| Report Form | Description |
|---|---|
| Initial form | Set up Format Control on the Initial form to initialize data entered on the Initial form. Use initialization expressions to validate data entered on the Initial form.<br>The Format Control record looks like Format Control on any ServiceCenter form. |
| Header form | Set up Format Control on the header form to initialize the variables, defined in the initialization expressions section of the Format Control record, for other forms to use. Use the header form to initialize a variable only once when executing the report. |
| Detail form | Set up Format Control on the detail form to initialize or calculate data, select records from secondary files, and run P4 subroutines, using the display option of the Format Control record. Validation expressions have no effect in this version of the Report Writer.<br>ServiceCenter resets all variables set in the detail form each time a user selects a record. |

## Accessing Format Control

Accessing Format Control through Forms Designer 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.

# How do I access Format Control?

### To access Format Control

1 When in the Report Maintenance form or Report Totals form, open **Options** > **Forms Designer**.
The initial Forms Designer view opens.

2 Type the name of the form you want to see in the **Form** field of the **Forms Designer** dialog box.
- or -
Click **Search** to find the form.

3 Open **Options** > **Format Control**.
The Format Control record appears for the selected form.

**Important:** Ensure that this Format Control record has the same name as the detail form.

# Format Control options

| Option | Description |
|---|---|
| Add | Adds the Format Control record. |
| Update | Updates the Format Control record. |
| End | Saves and Exits the Format Control record. |
| Calculations | Accesses the calculations section of the Format Control record.<br>Example calculation expression:<br>`if (flag=false)`<br>`then ($demo.elapsed=close.time in $file - open.time in $file)`<br>`else ($demo.elapsed=tod() - open.time in $file)` |
| More | Accesses other sections of the Format Control record. |
| End | Exits the Format Control record without saving the information. Returns back to the Report Writer Entry form. |

# Sample Report 3: Using Format Control for Calculations

This example report demonstrates how to define the necessary Totals Parameters to use calculations. You can use the calculations process within Format Control whenever you want to perform a calculation on currently available fields or variables.

### To use calculations in a report

1 Create the report. For this example, use the following parameters.

| Field | Value |
|---|---|
| Report Name | problem.calculations.demo |
| Report Title | Demo of Using Totals Calculations |
| Auto/Manual | auto |
| Summary/Detail | detail |
| Primary File | probsummary |
| Header Format Name | problem.calculations.demo.header |
| Detail Format Name | problem.calculations.demo.detail |
| Stacked Query | `flag=true and status~="closed"` |
| Sequence of Report | assignment |

For instructions on creating a report, see How do I create a new report? on page 16.

ServiceCenter checks for a form called *problem.calculations.demo* in the `format table`.

- If the form already exists, it opens.
- If the form does not exist, a blank screen opens, allowing you to fill in the form using normal Forms Designer procedures. For Forms Designer procedures, see the ServiceCenter online help.

2 Create the detail form using following input fields. For more information, refer to the ServiceCenter online help.

number
category
sassignment
status
open.time
$demo.elapsed

3 Add captions to identify the fields.

4 When you are finished adding the report, click **Edit Rpt** to access the edit functions. The report opens.

5 Click **Totals** to access the Report Totals form.

6 Type the name of the form being used to print the subtotal in the **Format Name** field.

For this example, type `problem.calculations.demo.subtotal`.

7 Create the subtotals form.

Open **Options** > **Forms Designer**.
If the form does not exist, a blank screen opens.

8 Fill in the detail form, following normal Forms Designer procedures. Open **Options** > **Forms Designer**. For more information, refer to the ServiceCenter online help.

For this example, create the form problem.category2.subtotal with the following values.

| Caption | Input |
|---|---|
| Summary totals For: | report.subfld |
| Average Elapsed Time: | $demo.elapsed.average |
| Total Elapsed Time: | $demo.elapsed.sum |

9 To calculate the average and sum of the variable `$demo.elapsed` for each grouping of problem records by category, type the values shown for the **Field/Variable Names to Total** and **Calc** fields.

| Field/Variable Names to Total | Calc |
|---|---|
| $demo.elapsed | AVG |
| $demo.elapsed | SUM |

10 Scroll left to view the second portion of the Report Totals form.

11 Specify values in the **Variable Names on Form** array to identify where to print the result of the calculation within the subtotals form.

| Field/Variable Names to Total | Calc | Variable Names on Form |
|---|---|---|
| $demo.elapsed | AVG | $demo.elapsed.average |
| $demo.elapsed | SUM | $demo.elapsed.sum |

12 Type `category` in the **Reset Field/Variable Name** field to calculate the totals separately for each category grouping.

13 When you are finished entering the appropriate data, click **End** to return to the Report Maintenance form.

14 To evaluate the value to be used for the `$demo.elapsed` variable, you must attach Format Control to the detail form.

a When you are finished with your design changes, open **Options** > **Format Control** to create a new Format Control record.

b A prompt asks you to save your changes. Click **Yes**.
The *formatctrl.maint.initial* form opens.

c Ensure the name of your Format Control record is the same name as your subheader form.

d Specify any initialization expressions, and click **Add**.
For this example, there are no entries required in the **Initialization Expressions** field.

e When you are finished, click **OK**.

f Click **Calculations** in the Format Control Maintenance form.

g Type the values in the fields as described below.

| Field | Value |
|---|---|
| Display | `true` |
| Calculation | `if (status in $file#"closed") then ($demo.elapsed=close.time in $file - open.time in $file) else (#$demo.elapsed=tod() - open.time in $file)`<br>Put all expressions on one expression line, separated by semicolons. Never place a semicolon at the end of the final expression. This causes ServiceCenter to expect another expression. |

h When you are finished, click **OK**. Continue to click **OK** until you return to the Report Maintenance form.

15 To run this report, follow the instruction in Running Reports on page 59.

# Sample report 4: Using Format Control for subheaders

You use the subheader form when designing a report to print the information that immediately precedes the detail information from a secondary file. The subheader form identifies the secondary file used to extract information for the detail form, therefore, you must define Format Control for the subheader form of all Subheader reports.

- The subheader form contains the fields from the primary file, defined in the report.
- The detail form contains fields from the secondary file, defined in the Secondary File Queries section of the Format Control record of the subheader form.
- A common field value connects the primary and secondary files.

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

### To use Format Control for subheaders

1 Create a new report in Report Writer, using the following values. For more information, see How do I create a new report? on page 16.

| Field | Value |
|---|---|
| Name | location.summary (or any unique name) |
| Title | Any title |
| Auto or Manual | Auto |
| Summary or Detail | Detail |
| Primary File | location |
| Header Format | location.summary.header (default) |
| Subhdr Format | location.summary.subhdr |
| Detail form | location.summary.detail (default) |
| Primary Query | true |
| Sort Sequence | location<br>Since stacked queries are not used, location must be a key in the primary file's dbdict before you can use it as a sort field. |

2 Create the header form.

a Open **Options** > **Forms Designer** and create a form in with the following input fields. For more information, refer to the ServiceCenter online help.

| Name |
|---|
| Location |
| Address |
| City |
| State |
| Zip |
| Primary Contact |

b Add captions to identify the fields.

c When you are finished working within the Forms Designer utility, click **OK**.

3 Create the subheader form.

a Open **Options** > **Forms Designer**. Create the form using the following values. For more information, refer to the ServiceCenter online help.

| Caption | Input |
|---|---|
| Location: | location |
| Location Name: | location.name |
| Address: | address |
| | window=1 |
| City: | city |
| State: | state |
| Zip: | zip |
| Primary Contact: | primary.contact |

b When you are finished working within the Forms Designer utility, click **OK**.

4 Add Format Control.

a Open **Options** > **Format Control** to create a new Format Control record. For more information, refer to the ServiceCenter online help.

b A prompt asks you to save your changes. Click **Yes**.
The *formatctrl.maint.initial* form opens

c Ensure the name of your Format Control record is the same name as your subheader form.

d Specify and initialization expressions and click **Add**.
In this example, there are no entries required in the **Initialization Expressions** field.

5 Add queries.

a Open **Options** > **Queries**.

b Click **Yes**.

c Make the file queries entries as needed.

For this example, use the following values.

| Field | Value |
|---|---|
| Display | true |
| Filename | operator<br>This identifies the operator table as the secondary file for this report. The *location* table and *operator* table are connected when the values in their **Location** fields match. |
| Query | location=location in $file |
| Req'd Query | false<br>This signifies that the match is not required for the subheader form to print. |

d Click **Save** to save the record.

e Click **OK** twice to exit the forms and return to the Report Maintenance form.

6 Create the detail form.

a From the Report Maintenance form, put your cursor in the **Detail** field of the Forms structure of the Report Maintenance form (on the location.summary.detail value) and open **Options** > **Forms Designer**.

ServiceCenter checks for a form called *location.summary.detail* in the `format` table. If the form already exists, it opens. If it does not exist, a blank screen opens, allowing you to paint the form using normal Forms Designer procedures. For more information, refer to the ServiceCenter online help.

b Click **Design** to create the form or change the existing form.

c Add the appropriate **Input** fields.
For this example, include:

| Caption | Input field |
|---|---|
| Logical Name: | logical.name |
| Serial Number: | serial.number |
| Manufacturer: | manufacturer |
| Model: | model |
| Type: | type |

d Click **Add** or **OK** to return to the Report Maintenance form.

7 To run this report, follow the instruction in Running Reports on page 59.

# Sample Report 5: 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 subtotals forms. 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 forms. they contain initialization and calculation expressions for processing record counts.

### To create a matrix report using Format Control

1 Use the following values to create the report.

| Field | Value |
|---|---|
| Name | problem.severity.summ (or any unique name) |
| Title | A user - defined title |
| Auto or Manual | Auto |
| Summary or Detail | Summary |
| Primary File | probsummary |

| Field | Value |
|---|---|
| Header Format | problem.severity.summ.header (default) |
| Detail format | problem.severity.summ.detail (default) |
| Primary Query | `flag=true`<br>Unix users can also use `status~="closed"`. |
| Sort Sequence | severity<br>To sort and break on the **Severity** field for a Summary report, the field must exist as a key in the *probsummary* dbdict, or the primary query must be defined as a stacked query. |

2 Create the header form.

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

ServiceCenter checks for a form called problem.severity.summ.header in the `format` table. If the form already exists, it opens. If it does not exist, a blank screen opens, which allows the user to paint the form using normal Forms Designer procedures. For more information, refer to the ServiceCenter online help.

b In order to sort and break on the **Severity** field for a Summary report, you must set up the same field as a key in the probsummary dbdict, or define the primary query as a stacked query.

c Click OK to add the new form and return to the Report Maintenance form.

3 Create a detail form.

Since this report is a Summary report, Report Writer does not print a detail form when it generates the report. However, because a Format Control record is required for calculations to occur on each detail record, we must create a detail form to attach to the Format Control record.

Place your cursor in the **Detail** field (on the problem.severity.summ.detail value) and open **Options** > **Forms Designer**.

ServiceCenter checks for a form called *problem.severity.summ.detail* in the `format` table. If the form already exists, it opens. If it does not exist, a blank screen opens, enabling you to paint the form using normal Forms Designer procedures.

4 Create the first subtotals form.

a Select **Options** > **Totals** to open the Report Totals form.

b Position the cursor on the first **Subtotals Format Name** field and open **Options** > **Forms Designer**.

c Type `problem.severity.summ.sub` and click **New.**

d A prompt asks if you want to use the Forms Designer wizard. Click **No**.

e Follow normal Forms Designer procedures to create a new form including the following input fields. For more information, refer to the ServiceCenter online help.

| | | |
|---|---|---|
| report.subfield | $open | $updt |
| $alrt1 | $alrt2 | $alrt3 |
| report.subcount,1 | | |

f Add captions to identify the fields.

g When you are finished, click **OK**.

5 Edit the Report Totals form.

a Open **Options** > **Totals**.

b Fill in the fields in the Report Totals form as follows:

**Subtotals Format Name** field: `problem.severity.summ.sub`

| Field/Variable Names to Total | Calc | Variable names on form |
|---|---|---|
| $opn | SUM | $open |
| $upd | SUM | $updt |
| $al1 | SUM | $alrt1 |

| Field/Variable Names to Total | Calc | Variable names on form |
|---|---|---|
| $al2 | SUM | $alrt2 |
| $al3 | SUM | $alrt3 |

c Access the second portion of the Report Totals form and scroll to the right until you see the **Reset Field/Variable Name** field.

d Type `severity`.

e Type `true` in the **ALWAYS** field.

f Click **OK**.

6 Create the second subtotals form.

a Position the cursor on the second **Format Name** field and type `problem.severity.summ.tot.`

b Open **Options** > **Forms Designer**.

c Follow normal Forms Designer procedures to create a new form with the following input fields. For more information, refer to the ServiceCenter online help.

| | | |
|---|---|---|
| open.tot | $upd.tot | |
| $alrt1.tot | $alrt2.tot | $alrt3.tot |
| report.count | | |

d Add captions to identify the fields.

e When you are finished, click **OK**.

7 Edit the Report Totals form.

a Open **Options** > **T**otals.

b Fill in the fields in the Report Totals form to add the following values.

**Subtotals Format Name** field: `problem.severity.summ.tot`

| Field/Variable Names to Total | Calc |
|---|---|
| $opn | SUM |
| $upd | SUM |
| $al1 | SUM |
| $al2 | SUM |
| $al3 | SUM |

c Access the second portion of the Report Totals form and scroll to the right until you see the **Reset Field/Variable Name** field.

d Type `severity`.

e Type `true` in the **ALWAYS** field.

f When you are finished, click **OK**.

8 Create a Format Control record for the header form.

a Position the cursor on the **Format Header** field and open **Options** > **Forms Designer**.

b Open **Options** > **Format Control**.

c The Format Control utility opens, enabling you to add a new Format Control record.

**Note:** Ensure that the name of this Format Control record is the same name as your header form.

d Type the following statements in the **Initializations Expressions** fields.

```
$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 run only once during the report generation.

e When you are finished, click **OK**. Continue to click **OK** until you return to the Report Maintenance form.

9 Create a Format Control record for the detail form.

a Position the cursor on the **Format Detail** field and open **Options** > **Forms Designer**.

b Create the necessary fields using the normal Forms Designer procedures. For more information, refer to the ServiceCenter online help.

c Click **OK**.

d Open **Options** > **Format Control**.

e When prompted, click **Yes**.

Ensure that the name of this Format Control record is the same name as your detail form.

f Type the following statements in the **Initializations Expressions** field.

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

**Note:** These initialization expressions run each time you select a new detail record.

g When you are finished, click **OK**.

h Click **Calculations** in the Format Control Maintenance form. The Format Control Maintenance - Calculations form opens.

i Type the values in the fields as described below.

| Field | Value |
|---|---|
| Display | `true` |
| Calculation | `if status in $file#"open" then $opn=1 ;`<br>`if status in $file#"upd" then $upd=1 ;`<br>`if (index("1",status in $file)> 0) then $al1=1 ;`<br>`if (index("2",status in $file)> 0) then $al2=1 ;`<br>`if (index("3",status in $file)> 0) then $al3=1`<br>Put all expressions on one expression line, separated by semicolons. Never place a semicolon at the end of the final expression. This causes ServiceCenter to expect another expression. |

j When you are finished, click **OK**. Continue to click **OK** until you return to the Report Maintenance form.

10 To run this report, follow the instruction in Running Reports on page 59.

# CHAPTER 4 Using Stacked Queries

Report Writer enables you to define selection, totaling, and report form criteria
Using 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.

## When should I choose the Primary Query option?

Choose the Primary Query option in the following circumstances:

- A query of *true* is defined and either:
  - No sort fields are defined (Report Writer uses the first key in the key list to retrieve the data records).
    - or -
  - The sort fields are defined as keys.
  - 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.

# Rules for primary queries

The following rules apply if you run a report with a primary query.

- Report Writer does not create a work table.
- For optimum performance, the primary query must be filly keyed.
- If you define sort fields, Report Writer uses that key (and that key only) to retrieve records from the database.

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

For example, if a table has the following keys,

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

and the sort fields for the report are,

- `open.time`
- `assignment`
- `category`

then the query,
`category="network"and status~="closed"`
is not fully keyed.

This query is partially keyed. The sort fields force ServiceCenter to use the second key to retrieve the records. Report Writer must expand each record the query retrieves (in this case all records appearing in the key list for the second key) to determine if the data record meets the search requirement.

# When should I choose the Stacked Queries option?

Stacked queries are separate and distinct from secondary queries.

Choose the Stacked Queries option in 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.

## Rules for stacked queries

The following rules apply if you run a report with stacked queries:

- Repertories creates a work table.
- Any Format Control definitions specified for any forms used in the report are still in effect.
- Any subtotaling requirements defined for the report are still in effect.
- Sort fields do not have to be keys in the primary file's dbdict, but they must exist as fields.
- If you use structure names in one sort field, you must use them in all sort fields.
- Report Writer performs the following validations on all sort fields and logs any errors 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.

**Important:** Do not specify other field types. Other types of fields can cause Report Writer to abort the sort.

- Report Writer runs each stacked query against the primary file and stores all records in the sort work table, temporarily.

- Report Writer runs the primary query against the sort work table, and therefore, under normal circumstances, is defined as true.
- The work table is a logical file defined with the following naming standard:
  `workxxxmmddyyhhmmss`
  where `work` is a constant, `xxx` is the name of the primary file and `mmddyyhhmmss` is the date and time that the work table was created.
- The work table is created in physical file 3. Exercise caution when using stacked queries. Stacked quires causes all records from any table to be read and placed in the sort work table.
- The work table, scheduled for deletion 2 hours after its creation, has a class of report and an operator name of sortsubr. Periodically review the dbdict file to ensure that all work tables have in fact been deleted. Be sure to only delete those files that completely follow the naming standards of the sort work tables.
- The sort routine copies all unique keys from the primary file to the sort work table and adds the sort fields as nulls&duplicates keys. All no nulls and no duplicates keys that appear in the keylist before the first unique key are copied to the sort work table. If no unique, no nulls or no duplicates keys exist, the first key in the keylist is copied to the sort work table.
- If the same record is retrieved using different queries and at least one unique key is defined for the sort work table, the duplicate record is not added to the sort work table.
- All error messages issued from the sort routine are logged in the msglog file.
- A message stating how many records were added to the sort work table based on the execution of each stacked query is issued to the msglog file.

# Stacked Queries Example

Suppose that the requirements of a report are:

1. List all open Incident tickets for the categories of lines, modems, and operator.

2. The sort requirements are by category, assignment, status, and open.time.

3 The primary file is *probsummary* and the following keys are defined.

```
number
**********************
flag
category
subcategory
company
********************
logical.name
********************
```

The specifications for the report would be:

| Field | Value |
|---|---|
| Primary File | probsummary |
| Primary Query | true |
| Stacked Queries | category="lines" and status=~"closed"<br>category="modems" and status=~"closed"<br>category="operator" and status=~"closed" |

**Note:** Each of these queries is fully keyed. For optimal performance, define the queries so that ServiceCenter can meet all or most of the query requirements from a key list rather than having to expand records retrieved from a particular key.

The sort fields would be:

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

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

If the report ran at 15: 00 hours on 02/07/06, the name of the sort work table would be *workprobsummary0207061500*.

# CHAPTER 5 Running Reports

You can run your report in the foreground, or schedule it for background processing.

## Running reports in foreground

When you run a report in the foreground the user's session and terminal are dedicated to the task until the report has completed.

### How do I run report in the foreground?

**To run a report in the foreground**

1. Access the report you want to run. See How do I access a report? on page 12.
2. When you have access to the report form, check the information to make sure it is correct.

   Click **Edit** to make any necessary modifications.
3. Click **Run** to run the report.

4 From the Run Report Exerciser form, you can 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 **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.
- If you change the default setting to false, Report Writer schedules the report to pass to the printer and print immediately after generating it.

c Click **View Spooled Reports**. The spoolheader form opens, which enables you to view the report output online.

d Click **End** to exit.

# Background Processing

When you schedule a report to run in the background, a simulated user session within the ServiceCenter system generates the report.

The default background processor (report), starts when ServiceCenter starts and processes all reports. Report Writer adds information about the report to the schedule table. The report background processor scans the schedule table every 60 seconds (the default scan interval), for any records whose scheduled time has elapsed and whose class is defined as report. When it finds a record, the background processor performs all functions necessary to generate the report to the spool and spoolheader tables. You can schedule reports to run in the future and specify a repeat interval. For example, you can schedule a report to run once every few hours, once a day, or once a week.

# How do I run a report in the background?

### To schedule a report for background processing

1 Access a report, following the procedures discussed in How do I access a report? on page 12. For this example, access the Problem Response Analysis by Assignment report.

2 From the Report Maintenance form, ensure that all the information about the report is correct. To modify any information, click **Edit** and make the necessary changes.

3 When you are finished editing the report, click **Run** to schedule the report for background processing.

The Run Report Exerciser form opens.

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.

6 Select one of the following print options.

Select (check) **Print** to print the report when it finishes spooling. Report Writer prints the report immediately after generating it.

Leave the field blank (unchecked) to only spool the report. The report remains in the spool table until you delete it schedule it to print.

- or -

Select (check) **Background** to print the report in the background when it finishes spooling. You must schedule the report to run at a particular time (defined in the next screen sequence) in a background processor.

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, see Running reports in foreground on page 59.

7 Type 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 in the report form.

8 Click **Run**.
The Schedule a Report (*report.prompt.sch1*) form opens.

9 Specify all the necessary values to generate the report.

10 Click **Schedule**.

ServiceCenter returns you to the Report Maintenance (report) form and displays the following message in the status bar: "Approx. time of exec: (00: 00: 00) for report: (name of report)."

A record in the schedule table now contains all the pertinent run information.

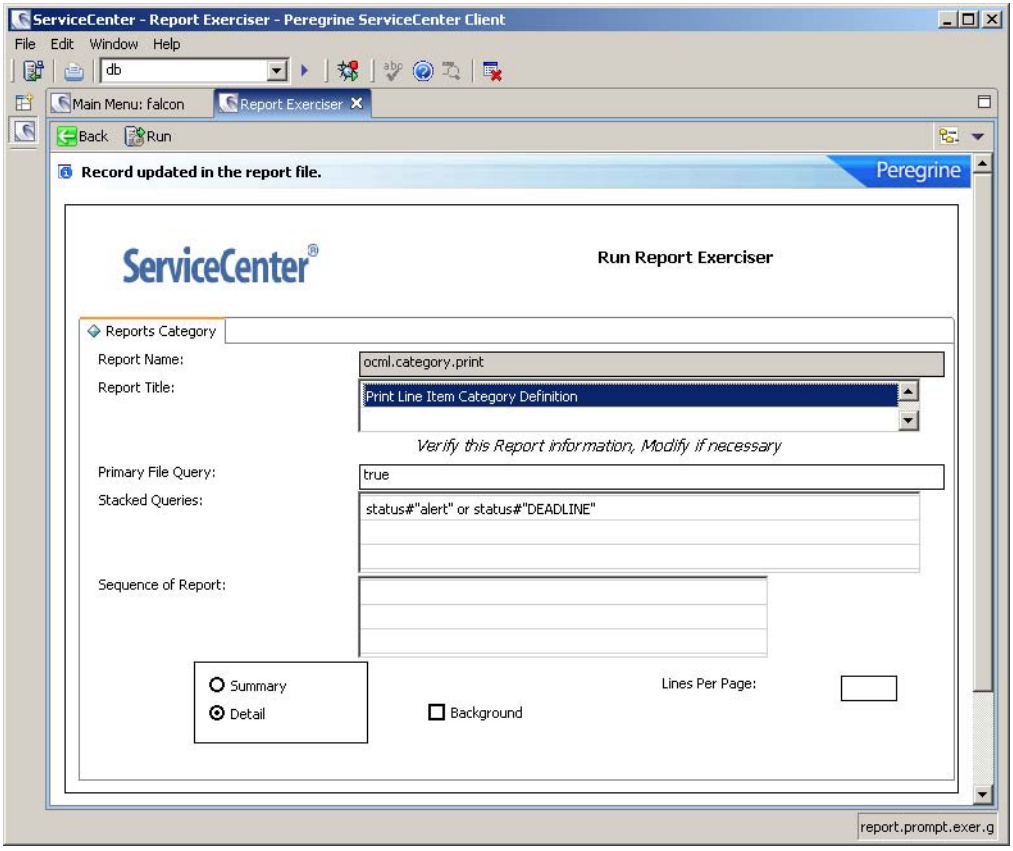
11 Press **Enter** to erase the message and return to Ready.

12 Open **Option** > **Maintenance Schedule** to view the schedule record. For more information, see Viewing and editing report schedules on page 64.

13 Open **Options** > **View Spooled Reports** to view the report. For more information, see How do I view a report online? on page 65.

## The Run Report Exerciser form

When run a report, Report Writer displays your options in the Run Report Exerciser form (report.prompt.exer).




The fields in this form are described in the ServiceCenter field help.

- To view the help for a field, put your cursor in the field and press **Ctrl+H** or select **Help** > **Help on Field**.

The available options may vary depending on the preferences you have selected, the record you select, and your permissions

| Button | Option | Description |
|---|---|---|
| | Run | Opens the Run Report Exerciser, enabling you to run a report in the foreground or background. It saves new reports and modifications made to existing reports. |
| | Back | Returns to the previously displayed form. |
| | Close Application | Closes the current application. |

# Viewing and editing report schedules

When you schedule a report for background processing, Report Writer adds it to the schedule table with a scheduled class of report. The schedule definition remains in effect until the background processor processes the report, or until you delete the record from the schedule table.

When the background processor begins processing the report, it changes the **Class** field from report to blank, and updates the **Status** field to application running. While both of these conditions exist, the background processor is actively processing the report.

If you specify a repeat interval in the schedule record, Report Writer uses the same schedule record 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.

## How do I edit report schedules?

### To view and edit a report schedule

1. Access a report that is currently scheduled to run, following the procedures discussed in How do I access a report? on page 12.
2. Select **Options** > **Maintenance Schedule**.
3. Make any necessary changes and click **Save**.

# Viewing reports

When Report Writer generates a report, it adds the report to the spoolheader and spool tables to be printed or available for online viewing. The spoolheader and spool tables link to each other. You cannot modify the contents of the report, but you can adjust the spoolheader table information, if necessary.

After sending a report to the spoolheader and spool tables, Report Writer passes it to the operating system based on parameters you define when scheduling the report to run.

The spoolheader table defines the name of the report, the time the report started and ended spooling, and the operator who scheduled the report to be processed.

The spool table contains the actual report (each page being a separate record within the spool table). You can view the generated reports online.

The fields in the spoolheader and spool forms are described in the ServiceCenter field help.

- To view the help for a field, put your cursor in the field and press **Ctrl+H** or select **Help** > **Help on Field**.

## How do I view a report online?

### To view a report online

1 Access a report, following the procedures discussed in How do I access a report? on page 12. For this example, access the resolution code listing report.

2 Select **Options** > **View Spooled Reports**.

Place your cursor in the field on which you would like to base your search, and click **Find**.

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

3 Click **Refresh** to refresh your view of the report.

4 Click **Back** to finish viewing the report.

PEREGRINE

# Index


**A**

accessing
- Forms Designer 27
- report totals 28
- Report Writer 12
- reports 12

**C**

control for subheaders, using format 42
copying reports 14, 15

**D**

deleting reports 15

**F**

format control
- accessing 37–38
- matrix report 46
- subheaders 42

format control for subheaders, using 42
forms
- detail 45
- initial 27

Forms Designer, accessing 27
forms, initial 27
forms, using query 21

**I**

initial forms 27

**M**

matrix report, format control 46

**O**

option, primary query 53
options
- copy/rename 14
- primary query 53
- stacked queries 55

**P**

primary query option 53

**Q**

queries 12
- primary 53
- stacked 55
- stacked, rules 55
- stacked, using 53, 55

queries, stacked 55
query forms, using 21
query option, primary 53

**R**

records, viewing report schedule 64
report schedule records, viewing 64
report totals, accessing 28
Report Writer, accessing 12
reports
- accessing report totals 28
- copying and renaming 14–15

creating and editing forms 20–36
deleting 15, 55
detail forms 45
forms 19, 37
grand totals 33
page breaks 28
processing in background 60
report totals 28
running in foreground 59
scheduling for background processing 61
summary totals 33
using calculations in 39–42
using grand totals in 33
using simple subtotals in 31
using subtotals in 33
using summary totals in 33
viewing online 65
reports, copying 14, 15
reports, deleting 15

## S

schedule records, viewing report 64
spool table 65
spoolheader table 65
stacked queries 55
using 53, 55
subheaders, using format control for 42

## T

table, spool 65
table, spoolheader 65
totals section, accessing 49

## U

using format control for subheaders 42
using query forms 21

## V

viewing report schedule records 64