CentreVu® Call Management System

Custom Reports

585-215-822
Comcode 108501867
Issue 2
December 1996

Copyright © 1996, 1999 Lucent Technologies
All Rights Reserved
Printed in U.S.A.

Notice
Every effort was made to ensure that the information in this book was complete and accurate at the time of printing. However, information is subject to change.

Your Responsibility for Your System's Security
Toll fraud is the unauthorized use of your telecommunications system by an unauthorized party, for example, persons other than your company's employees, agents, subcontractors, or persons working on your company's behalf. Note that there may be a risk of toll fraud associated with your telecommunications system and, if toll fraud occurs, it can result in substantial additional charges for your telecommunications services.

You and your system manager are responsible for the security of your system, such as programming and configuring your equipment to prevent unauthorized use. The system manager is also responsible for reading all installation, instruction, and system administration documents provided with this product in order to fully understand the features that can introduce risk of toll fraud and the steps that can be taken to reduce that risk. Lucent Technologies does not warrant that this product is immune from or will prevent unauthorized use of common-carrier telecommunication services or facilities accessed through or connected to it. Lucent Technologies will not be responsible for any charges that result from such unauthorized use.

Lucent Technologies Fraud Intervention
If you *suspect that you are being victimized* by toll fraud and you need technical support or assistance, call Technical Service Center Toll Fraud Intervention Hotline at 1 800 643-2353.

Trademarks
CentreVu is a trademark of Lucent Technologies.
DEFINITY is a registered trademark of Lucent Technologies.
INFORMIX is a registered trademark of Informix Software, Inc.
SPARC trademarks, including SPARCserver are trademarks or registered trademarks of SPARC International Inc. SPARCserver is licensed exclusively to Sun Microsystems, Inc. Products bearing SPARC trademarks are based upon an acrhitecture developed by Sun Microsystems, Inc.
Sun, Sun Microsystems, Sun Microsystems Computer Corporation, the Sunlogo, the SMCC logo, SunLink, SunSelect, Solaris, and Solstice DiskSuite are trademarks or registered trademarks of Sun Microsystems, Inc.
UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Corporation.
All other product names mentioned herein are the trademarks of their respective owners.

Ordering Information
Call: Lucent Technologies Publications Center
Voice 1 800 457-1235 International Voice 317 361-5353
Fax 1 800 457-1764 International Fax 317 361-5355
Write: Lucent Technologies Publications Center
P.O. Box 4100
Crawfordsville, IN 47933
Order: Document No. 585-215-822
Comcode 108501867
Issue 2, December 1999

You can be placed on a Standing Order list for this and other documents you may need. Standing Order will enable you to automatically receive updated versions of individual documents or document sets, billed to account information that you provide. For more information on Standing Orders, or to be put on a list to receive future issues of this document, please contact the Lucent Technologies Publications Center.

Lucent Technologies National Customer Care Center
Lucent Technologies provides a telephone number for you to use to report problems or to ask questions about your call center. The support telephone number is 1-800-242-2121. For technical support, customers outside the United States should call their Lucent Technologies Representative or Distributor.

European Union Declaration of Conformity
Lucent Technologies Business Communications Systems declares that the equipment specified in this document conforms to the referenced European Union (EU) Directives and Harmonized Standards listed below:

EMC Directive	89/336/EEC
Low Voltage Directive	73/23/EEC

CE
The "CE" mark affixed to the equipment means that it conforms to the above Directives.

Heritage Statement
Lucent Technologies—formed as a result of AT&T's planned restructuring—designs, builds, and delivers a wide range of public and private networks, communication systems and software, consumer and business telephone systems, and microelectronics components. The world-renowned Bell Laboratories is the research and development arm for the company.

Comments
To comment on this document, return the comment card at the front of the document.

Acknowledgment
This document was prepared by Global Learning Solutions, Lucent Technologies, Denver, CO.

Intellectual property related to this product (including trademarks) and registered to Lucent Technologies Inc. has been transferred or licensed to Avaya Inc.

Any reference within the text to Lucent Technologies Inc. or Lucent should be interpreted as references to Avaya Inc. The exception is cross references to books published prior to April 1, 2001, which may retain their original Lucent titles.

Avaya Inc. formed as a result of Lucent's planned restructuring, designs builds and delivers voice, converged voice and data, customer relationship management, messaging, multi-service networking and structured cabling products and services. Avaya Labs is the research and development arm for the company.

CentreVu®
Call Management System

Custom Reports

Table of Contents

Preface P-1

General Information P-1
Audience P-1
Introduction P-1

Organization and Use of This Document P-2
Organization of Document P-2
Use of Document P-4

Conventions Used in This Document P-5

Terminology P-6

Related Documents P-8
CentreVu CMS Publications P-8
Other Useful Publications P-9

Introduction to CentreVu CMS Custom Reports 1-1

General Information 1-1

Custom Reports and the Screen Painter 1-2
What Is a Custom Report? 1-2
The Screen Painter 1-2

Tasks to Create a Custom Report 1-4

Reports That Cannot Be Customized 1-6

Database Items Not in Standard Reports 1-7

How CentreVu CMS Stores and Retrieves Data 1-7
How CentreVu CMS Stores Data 1-7
How CentreVu CMS Retrieves Data 1-11

Getting Started 2-1

General Information 2-1

Defining a Report's Name, Access, and Type 2-2
Step 1: Access the Report Select Window 2-2
Step 2: Assign a Name to the Report 2-3
Step 3: Enter a Report Description 2-4
Step 4: Define the User ID 2-5
Step 5: Define Access to the Report 2-5

Step 6: Define the Report as Real-Time or Historical 2-6
Step 7: Save the Report Name 2-7

Changing a Report's Access or Description 2-8

Deleting a Custom Report 2-9

Using Screen Painter Editing Tools 3-1

Accessing the Screen Painter 3-1
Special Properties of the Screen Painter 3-2
Moving the Cursor on the Screen Painter 3-2
The Screen Painter Size 3-3
Scrolling the Screen Painter 3-4
Properties of Secondary Windows. 3-4

Copying an Existing Report Design 3-4
Step 1: Access the Get Copy Window 3-6
Step 2: Enter a Report Name 3-6
Step 3: Enter a User ID 3-7
Step 4: Select a Report Type 3-7
Step 5: Verify the Report 3-7
Step 6: Copy the Report 3-7

Editing a Report with Blocks 3-8
Erasing a Block 3-9
Moving a Block 3-10
Copying a Block 3-13

Entering Report Text 3-13

Saving Your Work 3-14

Defining the Data for a Custom Report 4-1

Defining Fields for the Report Input Window 4-1
Step 1: Access the Define Input Window 4-2
Step 2: Define the Variable Name 4-2
Step 3: Enter a Field Prompt 4-4
Step 4: Define the Number of Field Columns and Rows 4-4
Step 5: Define the Field as Single Value or Range/List 4-4
Step 6: Provide a Default Value for the Field 4-5
Step 7: Select a Field Type. 4-5
Input Fields That Allow Pattern Matching 4-7
Step 8: Associate an ACD with the Variable Field 4-8
Step 9: Save a Variable Input Field 4-9
Defining the Order in Which Input Fields Appear 4-10
Changing Report Input Fields. 4-11

Defining Report Fields 4-11
What the x's, v's, and h's Mean in a Field 4-12
Step 1: Define the Position and Length of a Field 4-13
Step 2: Define the Field's Data Expression 4-14
Database Items 4-14

Constants 4-15
Calculations 4-16
Calculation names 4-16
Aggregate Functions 4-18
count(*) 4-20
Data from More Than One Table 4-20
Step 3: Define the Table(s) for Calculation Names 4-21
Step 4: Justify Data in the Field 4-21
Step 5: Define the Field Format 4-22
Step 6: Save the Field Definition 4-27
Changing a Field Definition 4-27

Defining Bars in a Report 4-27
What the X's, V's, and H's Mean in a Bar 4-28
Step 1: Define the Position and Length of a Bar 4-28
Step 2: Define the Bar's Data Expression 4-30
Step 3: Define the Table(s) for Calculation Names 4-30
Step 4: Define the Bar Direction 4-30
Step 5: Define the Bar Scale 4-31
Step 6: Define the Maximum Graph Value 4-33
Step 7: Define the First Threshold 4-34
Step 8: Define the Second Threshold 4-34
Step 9: Select Normal or Reversed Thresholds 4-35
Step 10: Save the Bar Definition 4-36
Changing a Bar Definition 4-36

Defining the Rows of Data for a Report 4-37
Step 1: Access the Row Search Window 4-37
Step 2: List the Report Input Variables 4-38
Step 3: Select a Row Search ID 4-39
Step 4: Select Tables 4-39
Step 5: Select Rows in the Table(s) 4-42
Basic "Where" Clause 4-42
"Where" Clause with Variable 4-44
Multiple "Where" Clauses 4-44
"Where" Clause with a Range/List Variable 4-45
"Where" Clause for Selecting Rows from an ACD 4-45
"Where" Clause for Excluding Rows of Data 4-45
Step 6: Select a Field/Bar Type 4-46
Step 7: Select a Sort Type 4-48
Step 8: Select a Sort Order for Data 4-49
Step 9: Save Your Row Search Conditions 4-51
Assign a Row Search ID to Report Field(s) and Bar(s) 4-51
Changing Row Search Conditions 4-55
Changing the Row Search-Field Assignment(s) 4-56

Defining Fields to Show Run Time/Date and User Inputs 4-56
Step 1: Access the Variable/ Time/Date Window 4-57
Step 2: Define the Display Type and Format 4-57
Step 3: Save the Var/Date/Time Field 4-60

Saving Your Work 4-60

Completing Your Custom Report 5-1

Highlighting Fields. 5-1

Defining Stationary (No-Scroll) Areas 5-4
Changing a No-Scroll Area. 5-6

Saving the Report Design 5-7

Testing the Report Design 5-8

Test Design Error Messages 5-10
Phase 1 Error Messages 5-10
Phase 2 Historical Error Codes 5-14
Phase 2 Real-Time Error Codes 5-19

Running Custom Reports 5-20

Advanced Report Design 6-1

Creating a Custom Data Table 6-1
Step 1: Access the CentreVu CMS Database in INFORMIX . . . 6-3
Step 2: Build the Table 6-4
Step 3: Add Data to the Table 6-11
Modifying a Table. 6-14
Adding a Column 6-14
Changing a Column 6-14
Deleting a Column 6-14
Changing Data in a Table 6-16
Adding Rows of Data to a Table 6-16
Changing Rows of Data in a Table. 6-17
Deleting Rows of Data From a Table 6-17

Including Forecast Data in a Custom Report 6-18

Including Exceptions Data in a Custom Report 6-21

Selecting Rows from More Than One Table 6-25

Alternative Row Search Conditions 6-29
Selecting Rows Based on a Range of Values 6-29
Using Apostrophes for Some Database Item Values 6-30
Using String-Value Database Items 6-31
Other Available Formats for “Where” Clauses 6-32

Repeating Aggregate Function Values 6-33

Database Items and Calculations A-1

General Information A-1
Reference A-1

Glossary GL-1

Index IN-1

Preface

General Information

Audience

This book is written for all *CentreVu*™ Call Management System (*CentreVu* CMS) Release 3 Version 5 (R3V5) administrators and split/skill supervisors who are interested in creating custom reports.

Introduction

The *CentreVu*™ *CMS R3V5 Custom Reports* (585-215-822) document was written primarily for the *CentreVu* CMS administrator who has access to all parts of the *CentreVu* CMS, and secondarily for administrators (such as split/skill supervisors) who have limited access to the *CentreVu* CMS.

Organization and Use of This Document

This document takes you through creating a custom report from beginning to end, starting with Chapter 2, "Getting Started" and ending with Chapter 6, "Advanced Report Design".

For a first-time user of the Custom Reports subsystem, start at the beginning of the document and read the Preface and Chapter 1, "Introduction to CentreVu CMS Custom Reports", to become familiar with the content and structure of the book.

Organization of Document

The following list describes the contents of each chapter and appendix in this document.

Chapter 1 **Introduction to CentreVu CMS Custom Reports**

Chapter 1 gives the following information:

- Defines a custom report, the Screen Painter, and how *CentreVu* CMS stores and retrieves data
- Gives the tasks for creating a custom report
- Lists the standard *CentreVu* CMS reports that cannot be customized
- Lists the *CentreVu* CMS database items that are not in any standard reports but can be used in custom reports.

Chapter 2 **Getting Started**

Chapter 2 shows you how to:

- Define a custom report's name, access, and type
- Change a custom report's access or description
- Delete a custom report.

Chapter 3 **Using Screen Painter Editing Tools**

Chapter 3 shows you how to:

- Access the Screen Painter
- Copy an existing report
- Edit a report with blocks
- Enter report text
- Save your work.

Chapter 4 **Defining the Data for a Custom Report**

Chapter 4 shows you how to:

- Define fields for the report input window
- Define report fields
- Define bars in a report
- Define the rows of data for a report
- Define fields on the report to show when the report was run and what items the report covers.

Chapter 5 **Completing Your Custom Report**

Chapter 5 shows you how to:

- Highlight fields
- Define no-scroll areas
- Save and test a report design
- Understand test design error messages
- Run a custom report.

Chapter 6 **Advanced Report Design**

Chapter 6 shows you how to:

- Create a custom data table
- Include forecast data in a custom report
- Select rows from more than one table
- Do other alternative row search conditions
- Repeat aggregate function values in historical reports.

Glossary Defines Custom Reports terms and some *CentreVu* CMS terms.

Note

For a description of the *CentreVu* CMS system, see the *CentreVu™ CMS R3V5 Administration* (585-215-820*)* document.

For examples and explanations of standard reports, see the *CentreVu™ CMS R3V5 Reports* (585-215-821) document.

If you have the Forecast feature of *CentreVu* CMS, refer to the *CentreVu™ CMS R3V5 Forecast* (585-215-825) document to learn how to use the Forecast subsystem and read the reports.

For a description of the *CentreVu* CMS database items, see the *CentreVu CMS R3V8 Database Items and Calculations* (585-210-939) document.

Use of Document

Use this document to step you through the process of creating CMS custom reports. It includes all of the information needed to learn how to create the reports.

Conventions Used in This Document

The following conventions are used consistently throughout this document:

Related document callouts	Includes the document number (if a publication) and the book title, in italic. For example, ... See the *CentreVu*™ *CMS R3V5 Custom Reports* (585-215-822) document for more information.
`Courier font`	Designates a word or phrase quoted directly from the *CentreVu* CMS system. For example, the `Report destination:` field. Also designates an *INFORMIX** table name. For example, the `csplit` table.
`Courier bold` font	Designates something you type into the system. For example, type an **`x`** next to `Printer`, `Terminal`, or `File`.
“Chapter Names”	Are always in quotes.
italic	Highlights references to file names and directories. For example, */cms*.
Subsystem References	Are always initial capitalized. For example, ... set access permissions for *CentreVu* CMS users from the User Permissions subsystem.
Window Titles	Are always initial capitalized.

* INFORMIX is a registered trademark of Informix Software, Inc.

Terminology

The following terms are used throughout this book. For additional CMS definitions, please refer to the Glossary.

Current Interval	Represents the current intrahour interval which can be 15, 30, or 60 minutes. The current interval is part of the real-time database.
Daily Data	Interval data that has been converted to a 1-day summary.
Exception Reports	Display occurrences of unusual call-handling events.
Historical Database	Contains intrahour records for up to 62 days in the past, daily records for up to 5 years in the past, and weekly/monthly records for up to 10 years for each CMS-measured agent, split/skill, trunk, trunk group, vector, and VDN. Exceptions, call records, agent trace, agent login/logout, and forecast records are stored by the number of records (as defined in Data Storage Allocation).
Historical Reports	Display past ACD data for various agent, split/skill, trunk, trunk group, vector, or VDN activities.
Monthly Data	Daily data that has been converted to a monthly summary.
Multiuser Mode	Any administered CMS user can log into CMS. Data continues to be collected if data collection is “on.”
Previous Interval	One intrahour interval. At the end of each intrahour interval, the contents of the current intrahour interval are copied to the previous intrahour interval portion of the real-time database.
Real-Time Database	Current and previous intrahour data on each CMS-measured agent, split/skill, trunk, trunk group, vector, and Vector Directory Number (VDN).
Real-Time Reports	Display current Automatic Call Distribution (ACD) activity on agents, splits/skills, trunks, trunk groups, vectors, and VDNs.

Single-User Mode	Only one person can log into CMS. Data continues to be collected if data collection is “on.” This mode is required to change some CMS administration.
Subsystem	Each CMS Main Menu selection (for example, Reports, Dictionary, System Setup, Exceptions, along with Timetable and Shortcut, are referred to as subsystems.)
Weekly Data	Daily data that has been converted to a weekly summary.

Related Documents

There are many documents that can help you use the *CentreVu* CMS software to its maximum capability. The most important ones are listed below.

CentreVu CMS Publications

The following documents, relevant to *CentreVu* CMS, have also been published.

- *CentreVu* ™ *Call Management System Release 3 Version 5 Administration* (585-215-820)
- *CentreVu* ™ *Call Management System Release 3 Version 5 Real-Time and Historical Reports* (585-215-821)
- *CentreVu* ™ *Call Management System Release 3 Version 5 Change Description* (585-215-823)
- *CentreVu* ™ *Call Management System Release 3 Version 5 External Call History Interface* (585-215-824)
- *CentreVu* ™ *Call Management System Release 3 Version 5 Forecast* (585-215-825)
- *CentreVu* ™ *Call Management System Release 3 Version 5 Upgrades and Migration* (585-215-826)
- *CentreVu* ™ *Call Management System Release 3 Version 5 Sun** *SPARCserver*† *Computers Installation and Maintenance* (585-215-827)
- *CentreVu* ™ *Call Management System Release 3 Version 5 Sun SPARCserver Computers Connectivity Diagram* (585-215-828)
- *CentreVu* ™ *Supervisor Version 5 User Guide* (585-215-829)
- *CentreVu* ™ *Supervisor Version 5 Installation and Getting Started* (585-215-830)
- *CentreVu* ™ *Supervisor Version 5 Report Designer* (585-215-831)
- *CentreVu* ™ *Supervisor and Report Designer Version 5 Change Description* (585-215-832)
- *CentreVu CMS R3V8 Database Items and Calculations* (585-210-939).

*Sun is a registered trademark of Sun Microsystems, Inc.
†SPARCserver is a trademark of Sun Microsystems, Inc.

Other Useful Publications

At times you may need to reference other documents. You need to know where the documentation that was sent with the following is kept:

- Switch
- Terminals
- Printers
- *UNIX*[*]/Solaris[†] system.

*UNIX is a registered trademark of Novell, Inc.
†Solaris is a registered trademark of Sun Microsystems, Inc.

1 Introduction to *CentreVu* CMS Custom Reports

General Information

This chapter provides the following *CentreVu*™ Call Management System (*CentreVu* CMS) information:

- Definition of a custom report
- Definition of the Screen Painter
- Tasks to create a custom report
- Standard *CentreVu* CMS reports that cannot be customized
- *CentreVu* CMS database items that are not in any standard reports, but can be used in custom reports
- How *CentreVu* CMS stores and retrieves data.

Custom Reports and the Screen Painter

What Is a Custom Report?

A custom report is a report that you create and design using the Custom Reports subsystem. Like standard *CentreVu* CMS reports, a custom report displays information about Automatic Call Distribution (ACD) activity in your call center. For a custom report, you determine what specific ACD information is displayed and how it is displayed.

Like standard reports, custom reports fall into two categories: real-time and historical. A single custom report can contain either real-time data or historical data, but **not** both.

Also, you run a custom report via the Custom Reports subsystem, **not** from the Reports subsystem.

The Screen Painter

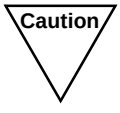
You design custom reports with a tool called the **Screen Painter**.

When creating *historical and real-time custom reports*, the Screen Painter tool lets you:

- Copy existing report designs, including standard report designs
- Edit reports using block moves, copies, and deletions
- Define ACD data for report fields
- Define ACD data for bar graphs (if you have the *CentreVu* CMS Graphics feature)
- Arrange report fields and bars
- Enter text for field labels, column headers, row headers, or special instructions
- Emphasize text and fields with highlighting options
- Define stationary (no-scroll) areas.

For *historical custom reports only*, the Screen Painter lets you:

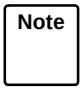
- Merge data in a report field to include data from different ACD entities (for example, defining a field that represents the percentage of calls an agent answered compared to all calls handled by that agent's split or skill)
- Merge data in a report field to include data with different time frames (for example, defining a field that represents the percentage of calls answered in an intrahour interval compared to all calls answered in the day)
- Include data from custom data tables that you create and populate within the *CentreVu* CMS database. If you want to create custom data tables. See Chapter 6, "Advanced Report Design".



If you create custom data tables, then you need to make sure there is enough disk space available to store the data. *CentreVu* CMS does not automatically check the available space. If you fill up your disk with custom data, then you will lose or damage stored custom and ACD data.

See the *CentreVu™ CMS R3V5 Administration* (585-215-820) document for more information about disk storage.

Do **not** tamper with standard ACD data in the *CentreVu* CMS database. If you do, then you will lose stored data.



You **cannot** merge real-time data with historical data.

Tasks to Create a Custom Report

There are 14 tasks you need to complete to create a custom report. If you consistently do **all** of these tasks for each report you create, then your reports will run properly and you will be able to create them efficiently. The 14 tasks are:

1. **Defining a Report's Name, Access, and Type (Chapter 2).**

   Define the *name* that you use both to run the report and to access the report design if you want to change the design. Define *access* to determine whether other users can run the report and copy the report's design to create their own custom reports. Define the *type* as either real-time or historical.

2. **Accessing the Screen Painter (Chapter 3).**

   Access the Screen Painter so that you can design the report. You can access the Screen Painter only after defining the report's name, access, and type.

3. **Copying an Existing Report Design (Chapter 3). (optional)**

   Copy an existing design so that you start with existing report headings, data fields, bars, and other report features. While this is an optional step, you normally save a lot of time and trouble by copying and modifying an *existing* report design. Both standard and custom report designs can be copied.

4. **Editing a Report with Blocks (Chapter 3).**

   Edit a report design using block moves, copies, and deletions. Editing with blocks lets you rearrange and delete sections of a report design quickly and easily. Block editing is particularly convenient when you have just copied an existing report design.

5. **Entering Report Text (Chapter 3).**

   Enter text to provide headings for the data fields in the report. Entering report text should be one of the first things you do so that:

   - The text provides a skeleton layout for positioning data fields.
   - The text helps you remember what data is supposed to go in the data fields.

   If you copy a report design the text of that report is copied. Using the Custom Reports subsystem, you can modify the text as required.

6. **Defining Fields for the Report Input Window (Chapter 4).**

   Define fields for the report input window so that users can run the report using parameters they choose (that is, what split, agent, time, date, and so on, the report is for).

   If you copy a report design, the definition of that report's input window is copied. You can then modify the input fields as required.

7. **Defining Report Fields and Defining Bars in a Report (Chapter 4).**

   Define the location of fields and bars in the report, as well as the field length, bar height/length, and the data that should appear in the fields/bars. Defining data consists primarily of specifying which database items supply data to the fields/bars. Your definition of report data is not complete, however, until you complete Task 8.

   If you copy a report design, the definition of that report's data fields/bars is copied. You can then modify the data fields/bars as required.

8. **Defining the Rows of Data for a Report (Chapter 4).**

   Define which rows of data in a table will supply data for the fields and bars you defined in Task 7.

   If you copy a report design, the definition of that report's rows of data will be copied. You can then modify the selection of rows as required.

9. **Defining Fields to Show Run Time/Date and User Inputs (Chapter 4).**

   Define fields on the report to show when the report was run and what items the report covers (as defined in the report input window).

   If you copy a report design these fields are copied. You can modify them as required.

10. **Highlighting Fields (Chapter 5).**

    Emphasize or de-emphasize individual fields and text in the report by changing brightness levels (or color if you have a color terminal) and by using underlines and reverse video.

    If you copy a report design, the highlighting and other video attributes will be copied. You can then modify them as required.

11. **Defining Stationary (No-Scroll) Areas (Chapter 5).**

    Define parts of the report that will stay in the same place in the report window even when you are scrolling up and down or right and left. You will normally define no-scroll areas for column headers, column totals, and row identifiers.

    If you copy a report design, no-scroll areas are copied. You can then modify them as required.

12. **Saving the Report Design (Chapter 5).**

    Save any work on the design of a custom report. Otherwise, any work you did will be lost.

13. **Testing the Report Design (Chapter 5).**

    Test your report immediately after designing and saving it. Testing helps eliminate wasted time in running a report whose design still has errors.

14. **Run the report.**

Reports That Cannot Be Customized

The following standard *CentreVu* CMS reports **cannot** be customized*:*

- Real-time: Multi-ACD report
- Real-time: Multi-ACD Top Agent report
- Historical: Split/Skill: Status report
- Historical: VDN: Status report
- Historical: System: Multi-ACD by Split/Skill report (daily, weekly, monthly)
- Historical: System: Multi-ACD Call Flow by VDN report
- Historical: Trunk Group: Busy Hour report
- Historical: VDN: Busy Hour report.

Database Items Not in Standard Reports

There are many more database items than are shown in the standard CMS reports. For a complete list and the definitions of all database items, see the *CentreVu CMS R3V8 Database Items and Calculations* (585-210-939) document.

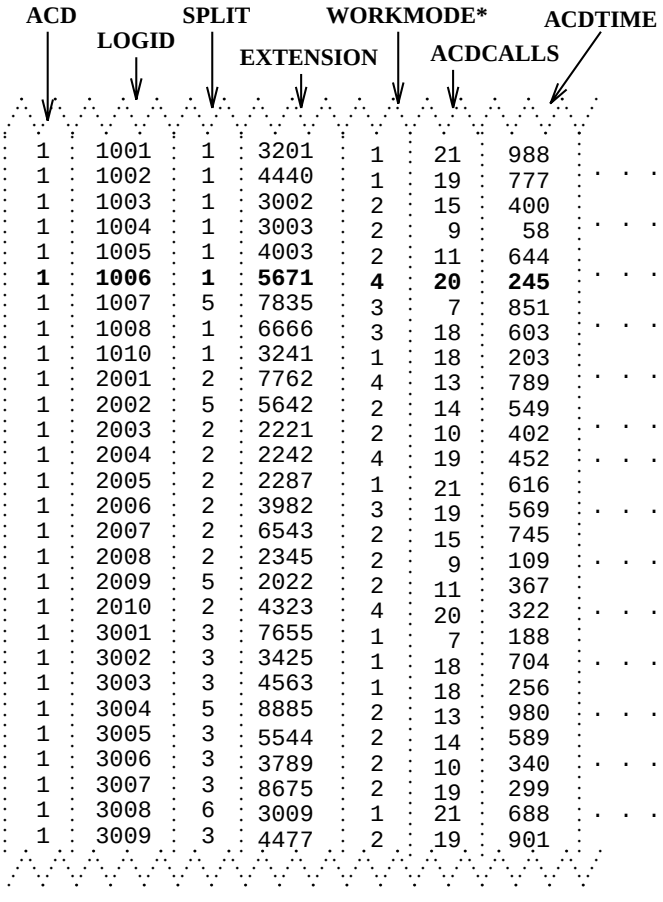
The *CentreVu™ CMS Release 3 Version 5 Real-Time and Historical* (585-215-821) document describes the database items associated with each real-time or historical report.

How *CentreVu* CMS Stores and Retrieves Data

The most important and difficult part of designing a custom report is defining the data that goes in the report. To define custom report data, you must first understand how *CentreVu* CMS stores and retrieves data.

How *CentreVu* CMS Stores Data

CentreVu CMS stores data in the *CentreVu* CMS database. The database is divided into 54 different tables that can be used in custom reports. A **table** is an array of columns and rows that stores data for a type of ACD element (split/skill, agent, trunk, trunk group, VDN, vector, call work code, forecasting, agent trace, call record, or exceptions) and for a specific time frame (for the current intrahour interval, for past intrahour intervals, for past day - summarized by day, and so on). Figure 1-1 shows how a small piece of a table (the Current Interval Agent table, in this case) might look in the database.



ACD	LOGID	SPLIT	EXTENSION	WORKMODE*	ACDCALLS	ACDTIME
1	1001	1	3201	1	21	988
1	1002	1	4440	1	19	777
1	1003	1	3002	2	15	400
1	1004	1	3003	2	9	58
1	1005	1	4003	2	11	644
1	**1006**	**1**	**5671**	**4**	**20**	**245**
1	1007	5	7835	3	7	851
1	1008	1	6666	3	18	603
1	1010	1	3241	1	18	203
1	2001	2	7762	4	13	789
1	2002	5	5642	2	14	549
1	2003	2	2221	2	10	402
1	2004	2	2242	4	19	452
1	2005	2	2287	1	21	616
1	2006	2	3982	3	19	569
1	2007	2	6543	2	15	745
1	2008	2	2345	2	9	109
1	2009	5	2022	2	11	367
1	2010	2	4323	4	20	322
1	3001	3	7655	1	7	188
1	3002	3	3425	1	18	704
1	3003	3	4563	1	18	256
1	3004	5	8885	2	13	980
1	3005	3	5544	2	14	589
1	3006	3	3789	2	10	340
1	3007	3	8675	2	19	299
1	3008	6	3009	1	21	688
1	3009	3	4477	2	19	901

* The numeric values for WORKMODE represent agent states that appear in reports. For example, 1 = AVAIL, 2 = ACD, 3 = ACW, and 4 = AUX.

Figure 1-1: Sample *CentreVu* CMS Table (Current Interval Agent Table)

Note

The example in Figure 1-1 shows data for the current intrahour interval for agents 1001 to 3009. Because data in this table is in real time, data changes second by second. The example represents a snapshot (or the most recent update) of the table.

Figure 1-2 shows how a small piece of the Historical Intrahour Interval Split table looks. (See the seventh line from the top of the table, 509 ACDCALLS.)

ROWDATE	STARTTIME	ACD	SPLIT	ACDCALLS	ABANDONS	ACDTIME	ABNTIME
070193	0800	1	1	443	48	36898	988
070193	0800	1	2	234	37	20012	777
070193	0800	1	3	111	20	13111	400
070193	0900	1	1	652	59	53442	1058
070193	0900	1	2	451	32	27635	644
070193	0900	1	3	93	11	15321	245
070193	1000	1	1	509	43	35401	851
070193	1000	1	2	391	31	19768	603
070193	1000	1	3	142	10	9786	203
070193	1100	1	1	480	39	33389	789
070193	1100	1	2	491	22	26789	549
070193	1100	1	3	297	15	12530	402
070293	0800	1	1	399	36	37651	1452
070293	0800	1	2	299	20	29602	7616
070293	0800	1	3	138	13	11523	2569
070293	0900	1	1	400	46	36178	1745
070293	0900	1	2	300	33	24303	1109
070293	0900	1	3	225	12	15628	367
070293	1000	1	1	394	40	40002	1322
070293	1000	1	2	323	34	29881	1188
070293	1000	1	3	105	14	12115	704
070293	1100	1	1	418	41	34819	1256
070293	1100	1	2	246	30	21173	980
070293	1100	1	3	100	18	10281	589
070393	0800	1	1	417	34	37856	1340
070393	0800	1	2	247	24	26308	1299
070393	0800	1	3	141	14	12567	688
070393	0900	1	1	444	43	39003	1001
070393	0900	1	2	301	31	27034	809
070393	0900	1	3	206	8	14230	445
070393	1000	1	1	420	51	39045	1733
070393	1000	1	2	299	39	29562	1303
070393	1000	1	3	198	24	12400	899
070393	1100	1	1	403	50	30990	1812
070393	1100	1	2	320	31	25410	904
070393	1100	1	3	99	21	10222	587

Figure 1-2: Sample *CentreVu* CMS Table (Historical Intrahour Interval Split Table)

Note

The example in Figure 1-2 shows data from July 1 to July 3, 1993, and simulates data for an ACD that has only three splits, 60-minute intrahour intervals, and activity each day from 8:00 a.m. to 12:00 p.m. only.

The *CentreVu* CMS database uses names to refer to columns of data in a table. These names are called **database items** in *CentreVu* CMS. In Figure 1-1 and Figure 1-2, database items are indicated with arrows pointing to their associated columns. The Current Interval Agent and Intrahour Interval Split tables actually contain many more columns (and hence many more database items) than are shown in the figures. For a complete listing of database items, see the *CentreVu CMS R3V8 Database Items and Calculations* (585-210-939) document.

Each **row** in a table contains data that is related by the value(s) of one or more of the columns. In Figure 1-1, each row in the Current Interval Agent table contains data related by agent login ID. If you look at the row for login ID 1006 (displayed in bold), you see that the agent is logged into Split 1 on extension 5671 and is currently in AUX work mode. In addition, up to this point in the current interval, the agent has had:

- 20 ACD calls (`ACDCALLS`)
- 245 seconds of ACD talk time (`ACDTIME`).

A column or a set of columns that cause the values in a row to be related is called an **index**. An index stores data sequentially and adds structure for the storage of data in the other columns. For each value in an index column, the remaining values in the corresponding row are related to that value. Thus, in Figure 1-1, the LOGID database item is an index.

In Figure 1-2, each row in the Intrahour Interval Split table contains data related by date, interval, and split. If you look at the row for Split 1 for the 10 o'clock interval on July 1, 1993 (displayed in bold), you see that Split 1 had:

- 509 ACD calls (`ACDCALLS`)
- 43 abandoned calls (`ABANDONS`)
- 35,401 cumulative seconds of ACD talk time for all ACD calls (`ACDTIME`)
- 851 cumulative seconds of wait time for all calls that abandoned before being answered (`ABNTIME`).

How *CentreVu* CMS Retrieves Data

CentreVu CMS retrieves data from the database based on three types of information you supply when you design a custom report:

- The name of the table(s)
- The database items in the table(s)
- The rows of data in the table(s).

For example, if you want a custom real-time agent report that lists the agents in a split, then the report design might appear as shown in Figure 1-3:

Note

Each series of `v's` in the illustration represents a report field for which it is expected that *CentreVu* CMS will find multiple values (in this case, values for more than one agent) and will display the values vertically in a column.

The `x's` by the `Split:` label represent the fact that only one value (in this case, a single split's name or number) is expected for the field.

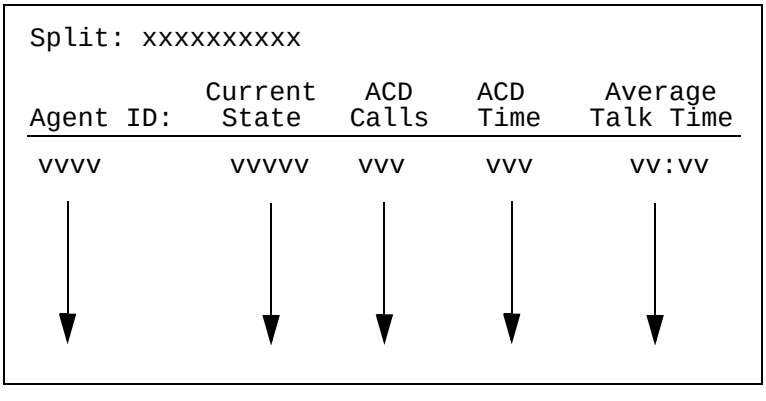


Figure 1-3: Sample 1 Custom Report Design

To tell *CentreVu* CMS how to retrieve data, you must tell *CentreVu* CMS to access the `cagent` (Current Interval Agent) table. Then, for each report field, you assign the appropriate database items. When you run the report, *CentreVu* CMS will find, in the `cagent` table, the columns of data associated with the database items (columns 2, 5, 6, and 7 in Figure 1-4).

Note

For the report field `Average Talk Time`, you actually assign a calculation, `ACDTIME/ACDCALLS`.

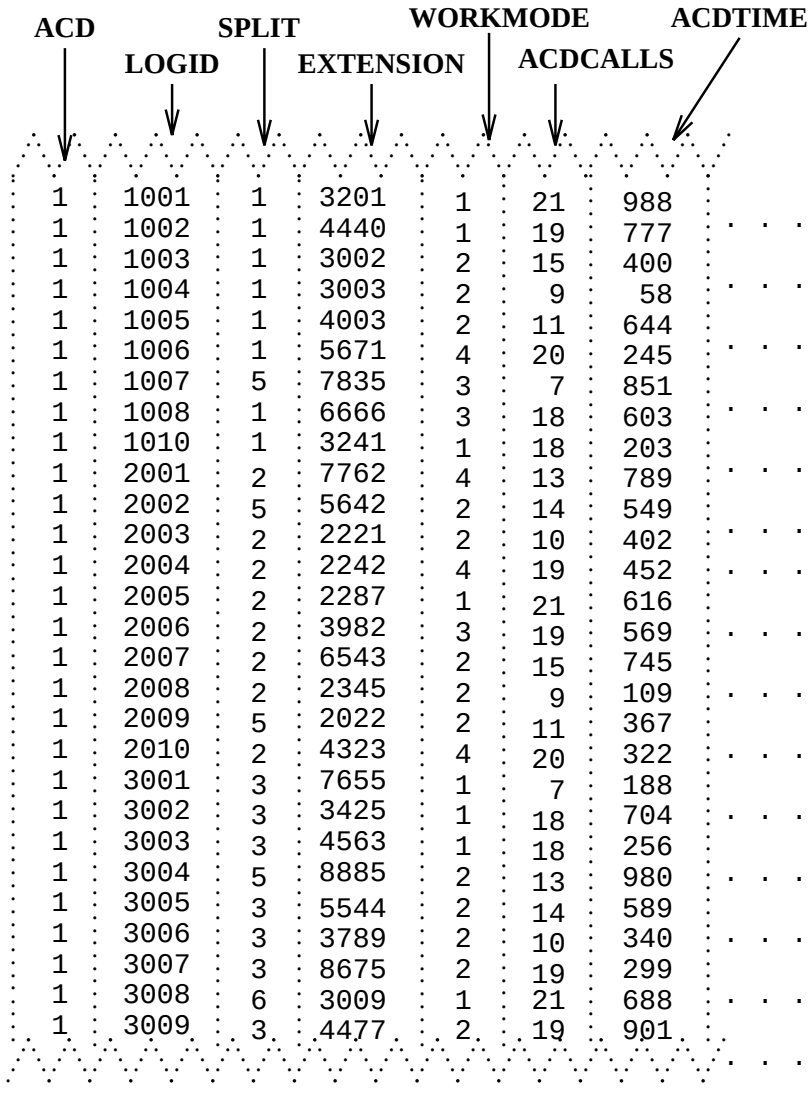
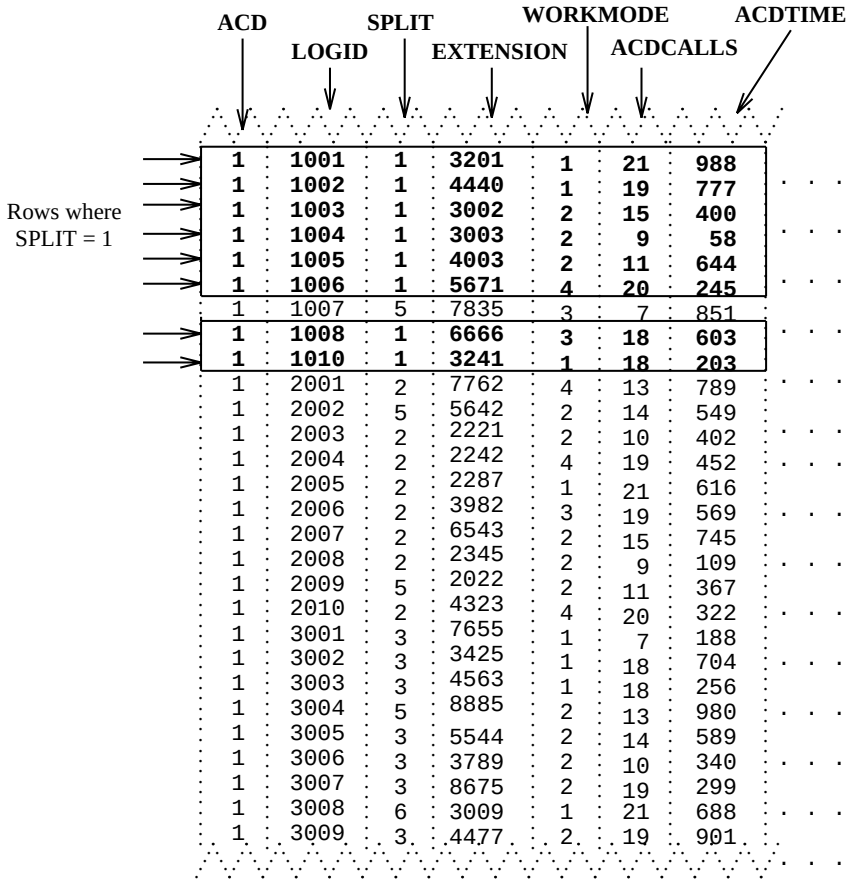


Figure 1-4: Sample 1 of Selection of Database Item

Next, you identify the appropriate rows that supply data. If you want agents in Split 1, you must tell *CentreVu* CMS to find rows that have the value 1 for the `SPLIT` database item. When you run the report, *CentreVu* CMS finds the appropriate rows of data in the `cagent` table (see rows with arrows in Figure 1-5).



ACD	LOGID	SPLIT	EXTENSION	WORKMODE	ACDCALLS	ACDTIME
1	**1001**	**1**	**3201**	**1**	**21**	**988**
1	**1002**	**1**	**4440**	**1**	**19**	**777**
1	**1003**	**1**	**3002**	**2**	**15**	**400**
1	**1004**	**1**	**3003**	**2**	**9**	**58**
1	**1005**	**1**	**4003**	**2**	**11**	**644**
1	**1006**	**1**	**5671**	**4**	**20**	**245**
1	1007	5	7835	3	7	851
1	**1008**	**1**	**6666**	**3**	**18**	**603**
1	**1010**	**1**	**3241**	**1**	**18**	**203**
1	2001	2	7762	4	13	789
1	2002	5	5642	2	14	549
1	2003	2	2221	2	10	402
1	2004	2	2242	4	19	452
1	2005	2	2287	1	21	616
1	2006	2	3982	3	19	569
1	2007	2	6543	2	15	745
1	2008	2	2345	2	9	109
1	2009	5	2022	2	11	367
1	2010	2	4323	4	20	322
1	3001	3	7655	1	7	188
1	3002	3	3425	1	18	704
1	3003	3	4563	1	18	256
1	3004	5	8885	2	13	980
1	3005	3	5544	2	14	589
1	3006	3	3789	2	10	340
1	3007	3	8675	2	19	299
1	3008	6	3009	1	21	688
1	3009	3	4477	2	19	901

Figure 1-5: Sample 1 of Selection of Table Rows

The data that *CentreVu* CMS plugs into the report is the data found in the intersection of the selected database items and rows. Therefore, the report shows data as shown in Figure 1-6:

Split: 1

Agent ID:	Current State	ACD Calls	ACD Time	Average Talk Time
1001	AVAIL	21	988	47:00
1002	AVAIL	19	777	40:09
1003	ACD	15	400	26:07
1004	ACD	9	58	6:44
1005	ACD	11	644	58:54
1006	AUX	20	245	12:25
1008	ACW	18	603	33:50
1010	AVAIL	18	203	11:28

Figure 1-6: Sample Custom Report 1

Note

Actually, when you design a custom report, you normally set up the row selection so that the users running the report can choose the rows in the report's input window. For example, to run the report in Figure 1-6, you would set up the row selection so users would fill out a Report Input window that asked them for a Split number. See Defining Fields for the Report Input Window in Chapter 4 of this manual for more information.

As mentioned earlier, *CentreVu* CMS uses indexes to create a structure for storing data. Similarly, *CentreVu* CMS uses these indexes to search for data. Indexes allow *CentreVu* CMS to find data much faster than if data were stored more randomly. Therefore, when you design a custom report, the rows of data for the report should be defined on the basis of index values. See "Defining Fields for the Report Input Window" in Chapter 4 of this manual.

Note

The indexes for each standard table are fixed and cannot be changed, deleted or added to. However, if you define a custom table in the *CentreVu* CMS database via *INFORMIX** SQL, you can define any indexes desired for that new table.

As another example of how *CentreVu* CMS retrieves report data is shown in Figure 1-7, which is an example of a report design for a custom intrahour interval split report that lists, by intrahour interval, data for a split in a single day.

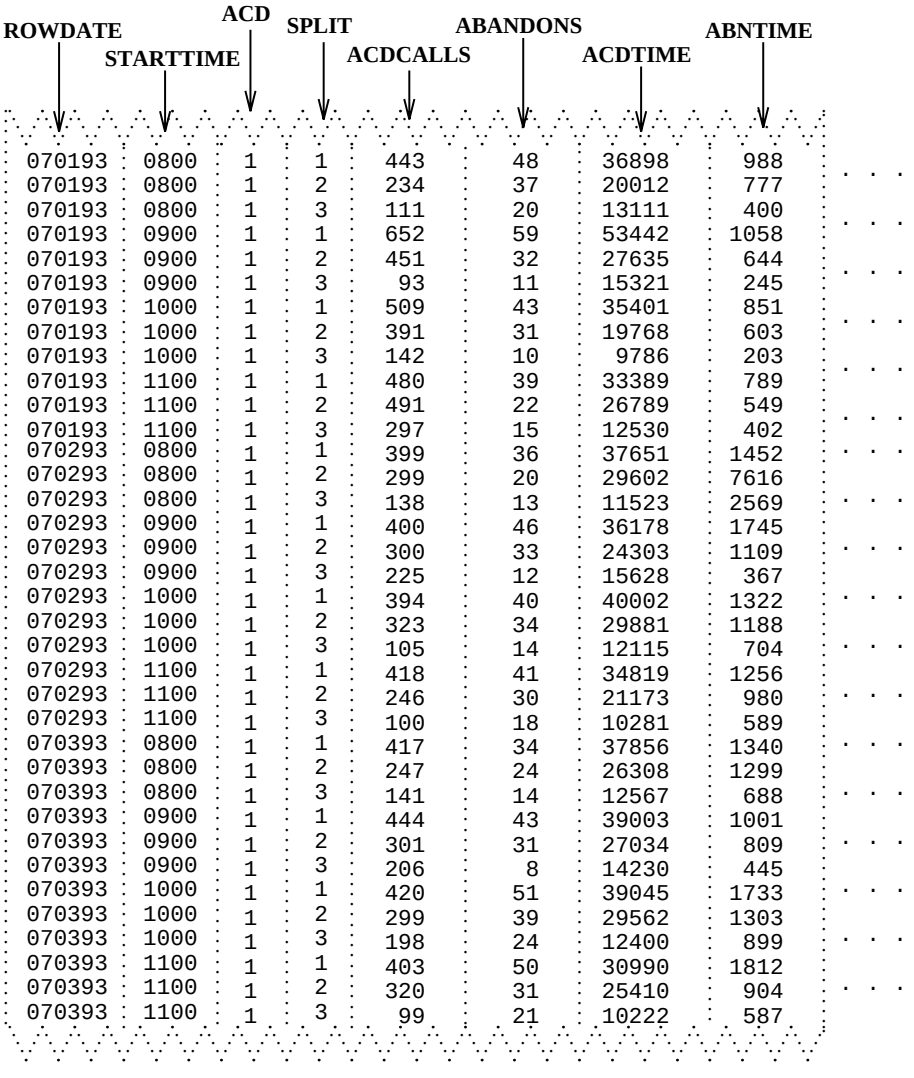
```
Split: xxxxxxxxxxx
 Date: xxxxxxxx

             ACD
Interval    Calls    Abandons
--------------------------------
 vvvvvvv     vvvv      vvvv
```

Figure 1-7: Sample 2 Custom Report Design

To tell *CentreVu* CMS how to retrieve data, you must tell *CentreVu* CMS to access the `hsplit` (Intrahour Interval Split) table. You must then assign the appropriate database items to the fields. When you run the report, *CentreVu* CMS finds the columns of data associated with the database items in the `hsplit` table (see Figure 1-8).

INFORMIX is a registered trademark of Informix Software, Inc.



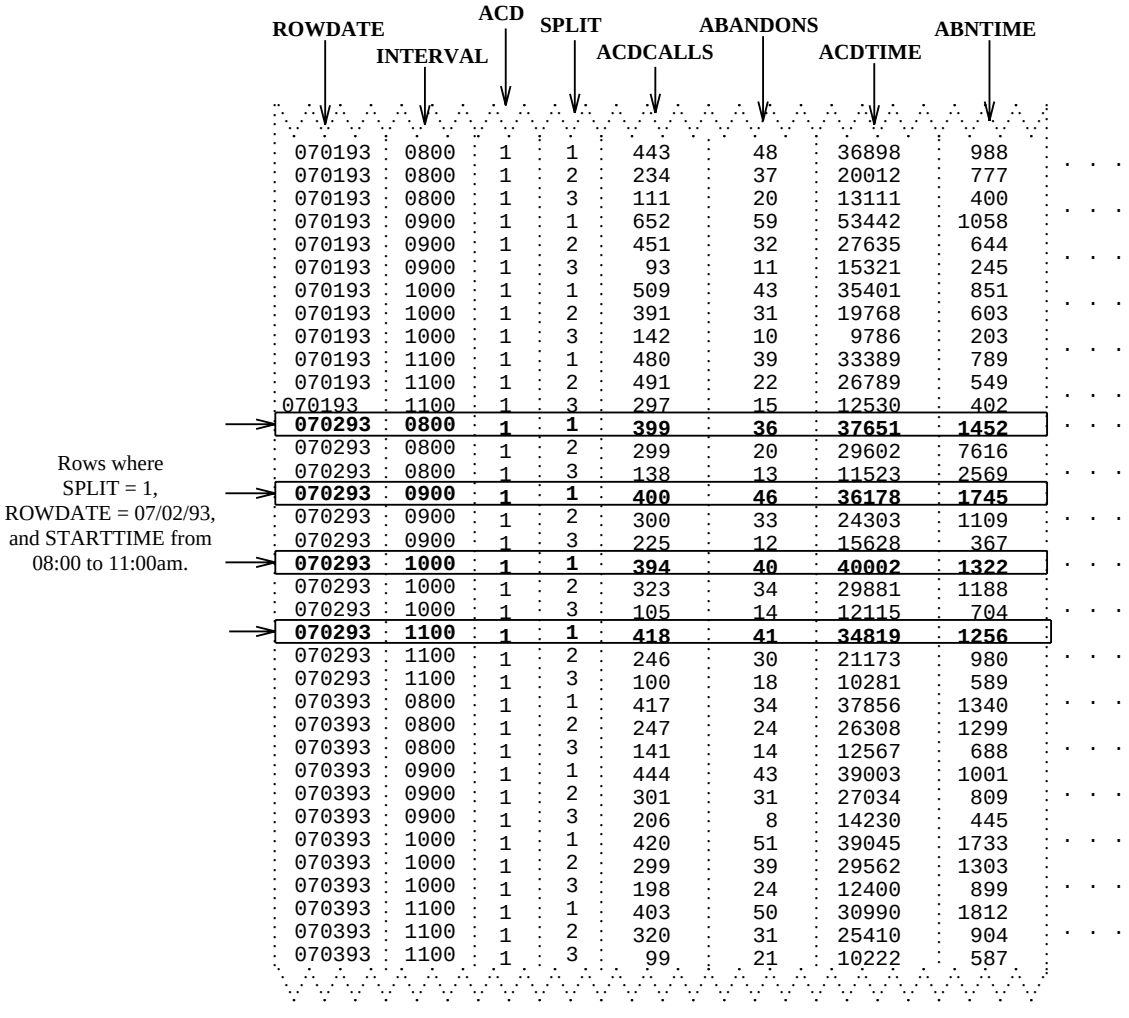
ROWDATE	STARTTIME	ACD	SPLIT	ACDCALLS	ABANDONS	ACDTIME	ABNTIME
070193	0800	1	1	443	48	36898	988
070193	0800	1	2	234	37	20012	777
070193	0800	1	3	111	20	13111	400
070193	0900	1	1	652	59	53442	1058
070193	0900	1	2	451	32	27635	644
070193	0900	1	3	93	11	15321	245
070193	1000	1	1	509	43	35401	851
070193	1000	1	2	391	31	19768	603
070193	1000	1	3	142	10	9786	203
070193	1100	1	1	480	39	33389	789
070193	1100	1	2	491	22	26789	549
070193	1100	1	3	297	15	12530	402
070293	0800	1	1	399	36	37651	1452
070293	0800	1	2	299	20	29602	7616
070293	0800	1	3	138	13	11523	2569
070293	0900	1	1	400	46	36178	1745
070293	0900	1	2	300	33	24303	1109
070293	0900	1	3	225	12	15628	367
070293	1000	1	1	394	40	40002	1322
070293	1000	1	2	323	34	29881	1188
070293	1000	1	3	105	14	12115	704
070293	1100	1	1	418	41	34819	1256
070293	1100	1	2	246	30	21173	980
070293	1100	1	3	100	18	10281	589
070393	0800	1	1	417	34	37856	1340
070393	0800	1	2	247	24	26308	1299
070393	0800	1	3	141	14	12567	688
070393	0900	1	1	444	43	39003	1001
070393	0900	1	2	301	31	27034	809
070393	0900	1	3	206	8	14230	445
070393	1000	1	1	420	51	39045	1733
070393	1000	1	2	299	39	29562	1303
070393	1000	1	3	198	24	12400	899
070393	1100	1	1	403	50	30990	1812
070393	1100	1	2	320	31	25410	904
070393	1100	1	3	99	21	10222	587

Figure 1-8: Sample 2 of Database Item Selection

Next, you must identify the appropriate rows that supply data. You might want data for the following:

- Split 1, which means you must identify rows that have the value 1 for the `SPLIT` database item.
- The date 07/02/93, which means you must identify rows with the value 070293 for the `ROWDATE` database item.
- The intrahour intervals 8:00 a.m. to 11:00 a.m., which means you must identify rows with the values 0800 through 1100 for the one database item.

CentreVu CMS then finds the appropriate rows of data (see the boxed rows in Figure 1-9).



ROWDATE	INTERVAL	ACD	SPLIT	ACDCALLS	ABANDONS	ACDTIME	ABNTIME
070193	0800	1	1	443	48	36898	988
070193	0800	1	2	234	37	20012	777
070193	0800	1	3	111	20	13111	400
070193	0900	1	1	652	59	53442	1058
070193	0900	1	2	451	32	27635	644
070193	0900	1	3	93	11	15321	245
070193	1000	1	1	509	43	35401	851
070193	1000	1	2	391	31	19768	603
070193	1000	1	3	142	10	9786	203
070193	1100	1	1	480	39	33389	789
070193	1100	1	2	491	22	26789	549
070193	1100	1	3	297	15	12530	402
070293	**0800**	**1**	**1**	**399**	**36**	**37651**	**1452**
070293	0800	1	2	299	20	29602	7616
070293	0800	1	3	138	13	11523	2569
070293	**0900**	**1**	**1**	**400**	**46**	**36178**	**1745**
070293	0900	1	2	300	33	24303	1109
070293	0900	1	3	225	12	15628	367
070293	**1000**	**1**	**1**	**394**	**40**	**40002**	**1322**
070293	1000	1	2	323	34	29881	1188
070293	1000	1	3	105	14	12115	704
070293	**1100**	**1**	**1**	**418**	**41**	**34819**	**1256**
070293	1100	1	2	246	30	21173	980
070293	1100	1	3	100	18	10281	589
070393	0800	1	1	417	34	37856	1340
070393	0800	1	2	247	24	26308	1299
070393	0800	1	3	141	14	12567	688
070393	0900	1	1	444	43	39003	1001
070393	0900	1	2	301	31	27034	809
070393	0900	1	3	206	8	14230	445
070393	1000	1	1	420	51	39045	1733
070393	1000	1	2	299	39	29562	1303
070393	1000	1	3	198	24	12400	899
070393	1100	1	1	403	50	30990	1812
070393	1100	1	2	320	31	25410	904
070393	1100	1	3	99	21	10222	587

Figure 1-9: Sample 2 of Selection of Table Rows

The data that *CentreVu* CMS plugs into the report is the data found in the intersection of the selected database items and columns. Thus, the report shows data as follows:

```
Split: 1
 Date: 07/02/93

             ACD
Interval    Calls    Abandons
-------------------------------
  8:00am      399        36
  9:00am      400        46
 10:00am      394        40
 11:00am      418        41
```

Figure 1-10: Sample Custom Report 2

Defining data is the central task of creating and designing a custom report. However, you must do many other tasks to create a custom report. Chapter 2, "Getting Started", walks you through these tasks.

2 Getting Started

General Information

This chapter tells you how to perform the following tasks in *CentreVu*™ Call Management System (*CentreVu* CMS):

- Define a custom report's name, access, and type (the **first** task in designing a custom report)
- Change a custom report's access or description
- Delete a custom report.

You do these tasks via the Edit Report: Report Select window.

To complete the design of a custom report, you must also follow the procedures in Chapter 3, Chapter 4, and Chapter 5, and, optionally, Chapter 6.

To create, change, or delete a custom report, you must have **read** and **write** permission for the Custom Reports subsystem.

Defining a Report's Name, Access, and Type

Use the steps in this section to define a new custom report's:

- Name
- Access
- Type.

These are the first steps in designing a new custom report.

Note

If you are a *CentreVu* CMS administrator, you can also specify the owner of the report you are defining (see "Step 4: Define the User ID" in this procedure). If you are not a *CentreVu* CMS administrator, you can define a report with only you as the owner.

You cannot define a custom report with a timetable.

Step 1: Access the Report Select Window

1a. Select the `Custom Reports` option on the Main Menu and press the Return key. → *The Custom Reports submenu displays.*

Figure 2-1: Custom Reports Main Menu

1b. Select the `Edit Reports` submenu option. → *The Report Select window displays.*

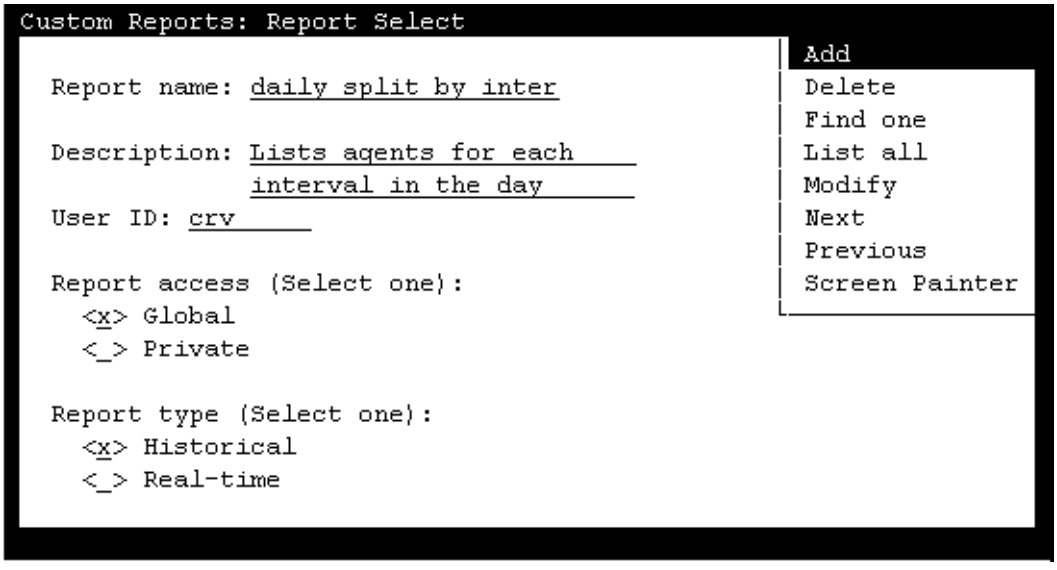


Figure 2-2: The Report Select Window

Step 2: Assign a Name to the Report

Enter a name for your report in the `Report name` field. The name can have up to 20 characters, including blanks. Because the name you give your report must be unique, you may want to look at existing custom report names before entering a name for your report.

Use the following steps to list existing report names.

2a. Clear the fields.

2b. Enter an **x** to select a report access—`global` or `private`.

2c. Enter an **x** to select a report type—`Historical` or `Real-time`.

2d. Press the Return key, select `List all`, and press the Return key again. → *The List All window displays all global or private custom report names for the specified report type.*

2d. Check the list to ensure that the name you want to give your report has not already been used. If you are creating a *global* report, its name must be different from any name in the list. If you are creating a *private* report, its name must be different from names of any existing global or private reports you have created.

Note
Your private report names can be the same as another user's private report names.

2e. Press the Exit key to return to the Report Select window → *The List All window disappears, and the cursor returns to the Report Select window.*

2f. Enter the name you want for your report.

Note
You can `List all` reports based on different combinations of field entries. You must always select a report type, `Historical` or `Real-time`.

Step 3: Enter a Report Description

If desired, enter a description of the report in the `Description` field. The description can have up to 50 characters, including blanks.

Your description should be detailed enough to describe the report's contents accurately.

Warning

Do not use the following characters:

\	backslash
;	semicolon
'	grave accent
~	tilde
"	double quotes
|	pipe
*	asterisk
?	question mark

Step 4: Define the User ID

Enter the user ID of the owner of the report. The default is your user ID. If you are a *CentreVu* CMS administrator, you can enter another user's ID if you wish to create a custom report for that user. This might be the case if you want only that user to be able to run the report.

Note

If you want to list another user's existing custom reports, you can enter that user's ID and do a `List All.` You do not have to be an administrator to list other users' custom reports.

Step 5: Define Access to the Report

Enter an `x` to select a report access option, either `global` or `private`.

Note

It is a good idea *initially* to make your reports private until they have been debugged and run successfully. This prevents the possibility of other users running reports you have saved but not yet tested.

`Global` access to the report gives other users the following capabilities:

- Other users can run the report.
- Other users can copy the report design when designing their own custom report (see "Copying an Existing Report Design" in Chapter 3 of this document).

If you select `global`, your report's name must be different from the name of any other custom report — global or private.

`Private` access to the report means that only you (and the *CentreVu* CMS administrator(s)) can run the report. In addition, no other users, except for *CentreVu* CMS administrators, can copy or run the report design for use in their own custom reports.

If you select `private`, your report's name must be different from the names of the following:

- Any other private report you have created of the same type
- Any user's global custom report of the same type.

Note

No other *CentreVu* CMS user other than a *CentreVu* CMS administrator can modify a report design you create, regardless of whether the report is global or private. A *CentreVu* CMS administrator always has the ability to modify your report design, even if you make it private.

Step 6: Define the Report as Real-Time or Historical

Enter an **`x`** to select a report type option, `Real-time` or `Historical`.

When you design the report on the Screen Painter, you are able to access data only for the category you select here. For example, if you select `Real-time`, you are not able to specify historical data in the report design.

Also, when you go to run the report, *CentreVu* CMS lists the report under the appropriate submenu, `Real-time` or `Historical`.

You **must** select the `Historical` option if you want to include:

- Exceptions data
- Forecast data
- Call work code data
- Agent trace data
- Call record data.

Step 7: Save the Report Name

1. Press the Return key, select `Add`, and press the Return key again.

→ *`Successful` appears on the status line. If CentreVu CMS will not add the report name because the name is not unique, `List all` to see what names already exist, and go back to "Step 2: Assign a Name to the Report" in this procedure.*

After you have added the report name, you can access the Screen Painter to begin designing the report (see Chapter 3).

Changing a Report's Access or Description

Note
You cannot change a report's name, type, or owner once you have added the report. Instead, you must create a new report with the desired name, type, or owner, and then copy the design of the old report to the new report.

1. On the Report Select window, complete the fields you wish to search on. (`Report type` is a required field `for` *CentreVu* `CMS`.) Press the **Return** key, and select `List all`. Press the **Return** key again.
   → *A List All window displays a list of all custom reports for the specified type and user ID.*

   Note
   You can change the `User ID` field to list another user's custom reports. You **cannot** change the user ID of a report that already exists.

2. Find the report you want to change, and press **Exit** to return to the Report Select window.
   → *The cursor returns to the previously selected field of the Report Select window.*

3. In the `Report name` field on the Report Select window, enter the name of the report whose description or access you want to change.

4. If necessary, change the default `Report type` selection. Then, press the **Return** key, select `Find one`, and press the **Return** key again.
   → *CentreVu CMS fills in all fields with the report's characteristics*.

5. Change the description or report access as desired. Then press the **Return** key, select `Modify`, and press the **Return** key again.
   → `Successful` *appears on the status line*.

   Note
   You cannot change a report's name or user ID using `Modify`. To change a report name or user ID, you must add a new report name (with the desired user ID), access the Screen Painter, copy the old report's design to the new report, and finally delete the old report name.

Deleting a Custom Report

Note
You cannot delete another user's custom reports unless you are a *CentreVu* CMS administrator.

1. On the Report Select window, complete the fields you wish to search on, and select `List all`. (`Report type` is a required field for *CentreVu* CMS.) → *A List All window displays a list of all global and private custom reports for the specified type and user ID.*

   Note
   You can change the `User ID` field to list another user's custom reports. However, you cannot delete another user's report unless you are a *CentreVu* CMS administrator.

2. In the List All window, find the report you want to delete, and press Exit to return to the Report Select window → *The List all window disappears, and the cursor returns to the first field of the Report Select window.*

3. In the `Report name` field on the Report Select window, enter the name of the report you want to delete.

4. Press the Return key, select `Find one`, and press the Return key again. → *CentreVu CMS fills in all fields with the report's characteristics*.

   Note
   If more than one report has the same name, you may have to use `Next` to find the report you want.

5. Press the Return key, select `Delete`, and press the Return key again. → `Successful` *appears on the status line.*

3 Using Screen Painter Editing Tools

Accessing the Screen Painter

To design your *CentreVu*™ Call Management System (*CentreVu* CMS) custom report, use the Screen Painter (Figure 3-1). On the Screen Painter, you can enter report text, data fields, and data bars in a layout that closely resembles the layout of the actual report.

You must define the report name and its select characteristics in the Edit Report: Report Select window before you can access the Screen Painter for that report.

Note

You can access the Screen Painter for a report that someone else designed only if you are a *CentreVu* CMS administrator. However, if another user's report has global access, you can add your own report name and then copy that user's report design on the Screen Painter (see the "Copying an Existing Report Design" section in this chapter).

Access the Screen Painter using the following steps.

1. On the Report Select window, enter the name of the report in the `Report name` field, and select `Find one`. → *The select characteristics of the report appear.*
2. Select the `Screen painter` action list option. → *The Screen Painter appears.*

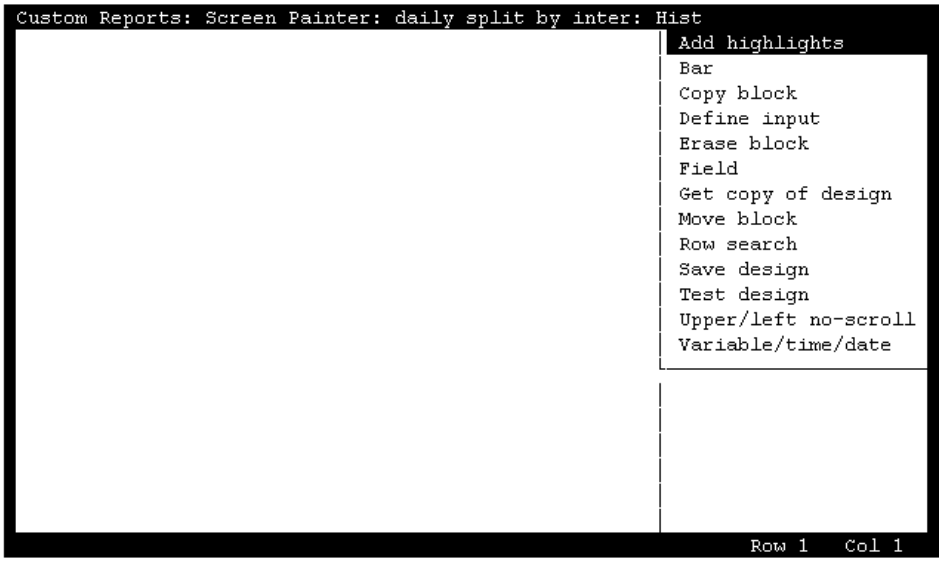


Figure 3-1: The Screen Painter

Special Properties of the Screen Painter

On the Screen Painter, cursor movement, scrolling, and data entry all differ from those operations in normal *CentreVu* CMS windows.

Moving the Cursor on the Screen Painter

Since the Screen Painter has no predefined fields, you can freely move the cursor around the interior of the Screen Painter with the following keys:

Arrow keys (↓, ↑, →, ←)	Move the cursor one space in the direction of the arrow.
Tab	Moves the cursor eight spaces to the right. You can also use Tab when you are defining blocks (see the "Editing a Report with Blocks" section in this chapter).
Shift Tab	This means you must **press and hold** the Shift key; then press the Tab key. This convention applies throughout this document when any set of two keys are shown side by side. Moves the cursor eight spaces to the left. You can also use Shift Tab when you are defining blocks (see the "Editing a Report with Blocks" section in this chapter). Shift Tab may not be available on some terminals.
Back Space	Moves the cursor one space to the left.
Ctrl f	(Forward) Moves the cursor to the far right edge (132nd column) of the Screen Painter.
Ctrl b	(Back) Moves the cursor to the left edge (1st column) of the Screen Painter.
Ctrl d	(Down) Moves the cursor to the bottom (Line 25) of the Screen Painter.
Ctrl u	(Up) Moves the cursor to the top (Line 1) of the Screen Painter.

The Screen Painter Size

The Screen Painter allows you to create a report design with maximum dimensions as follows:

- A horizontal size of 132 columns (character spaces)
- A vertical size of 25 lines.

For most terminals, *CentreVu* CMS displays (in the lower right border of the Screen Painter) the exact position of the cursor within the Screen Painter's 25 column by 132 column grid (see Figure 3-2). However, because the interior of the Screen Painter window is only 54 columns wide and 20 lines high, you may need to scroll the Screen Painter horizontally or vertically to access areas that do not fit within the window.

Note

The widest custom report that you can display on your terminal without scrolling is 78 columns wide. The terminal actually displays 80 columns, but the side borders of the report window use two of the columns.

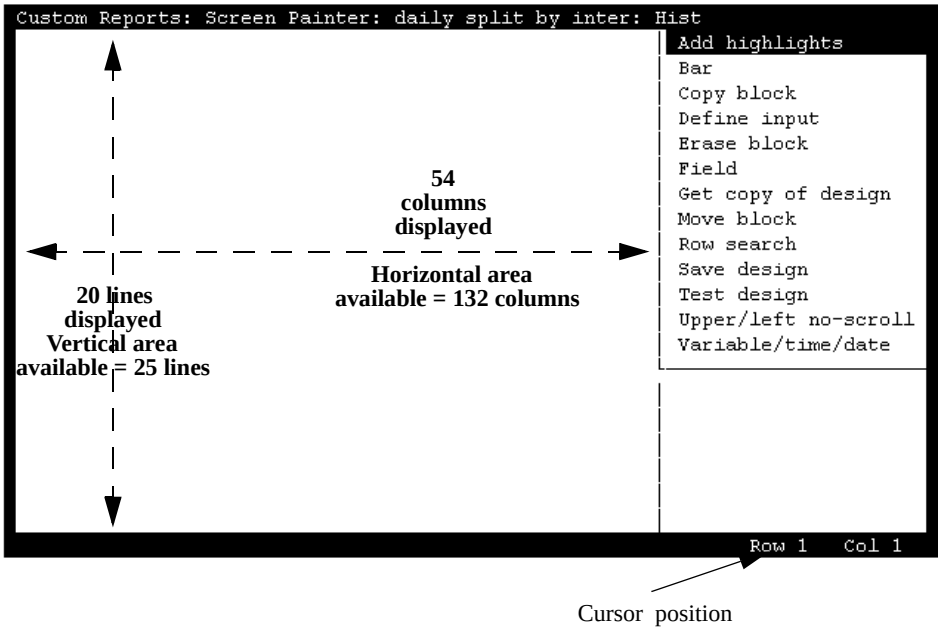


Figure 3-2: Screen Painter Size and Cursor Position

Scrolling the Screen Painter

The Screen Painter scrolls automatically when the cursor hits the left, right, top, or bottom border. The Screen Painter scrolls one column or one line at a time when you use the arrow keys, but scrolls eight columns at a time when you use Tab for horizontal scrolling. Right scrolling beeps when the 132nd column appears. Left scrolling beeps when the first column reappears. Down scrolling stops when the 25th line appears. Up scrolling stops when the top line reappears.

Properties of Secondary Windows

For most tasks on the Screen Painter, you must complete a secondary window. These secondary windows pop up after you select an action list option. Some secondary windows automatically disappear when you are finished entering data and select an action list option. However, for some secondary windows, you must press the Exit Screen Labeled Key (SLK) to close the window and return to the Screen Painter.

If you use the Current SLK to leave a secondary window and return to the Screen Painter, the secondary window remains open. However, until you close the secondary window, the Screen Painter is locked and you cannot enter text or select any other action list options.

Editing Keys in Secondary Windows

In the secondary windows only, you can use the standard field editing keys. These keys, which offer you considerable convenience, are as follows:

Ctrl e	Turns insert mode on or off. With insert mode on, you insert characters at the cursor's current position in the field.
Ctrl x	Erases characters in a field from the current position to the end of the field.
Ctrl y	Erases all characters in a field.
Ctrl z	Erases inputs in all fields in a window.

Note

These keys are **not** available on the Screen Painter itself.

Copying an Existing Report Design

In the majority of cases, you should begin your design of a custom report by copying an existing report design, then making desired changes. Copying,

then modifying, an existing report design is usually the quickest, easiest method for designing a custom report.

You can copy standard report designs or custom report designs. You can also copy more than one report design into a single custom report. However, if your report is a historical report, you cannot copy a real-time report onto the Screen Painter. Likewise, if your report is a real-time report, you cannot copy a historical report onto the Screen Painter.

Caution

Run test design on a report copy before modifying it. This ensures the copy is working before you do any modification to it.

Note

Items in the report you are copying may overlap text, fields, or bars you previously entered on the Screen Painter. If items in the copied report overlap existing items, *CentreVu* CMS copies to the Screen Painter only those parts of the report that **do not** overlap.

To prevent overlapping, you may need to clear the area in the upper left portion of the Screen Painter before copying a report design. You can clear the area easily by moving or deleting all data as a block (see the "Editing a Report with Blocks" section in this chapter). Be sure the cleared space is large enough to contain the report design you are copying.

Note

You **cannot** copy the following standard reports:

- Real-time: Multi-ACD report
- Real-time: Multi-ACD Top Agent report
- Historical: Split/Skill: Status report
- Historical: VDN: Status report
- Historical: System: Multi-ACD by Split/Skill report
- Historical: System: Multi-ACD Call Flow by VDN report
- Historical: Trunk Group: Busy Hour report
- Historical: VDN: Busy Hour report.

To copy an existing report design, do the following:

Step 1: Access the Get Copy Window

On the Screen Painter, select the `Get copy of design` action list option. → *The Get Copy window appears.*

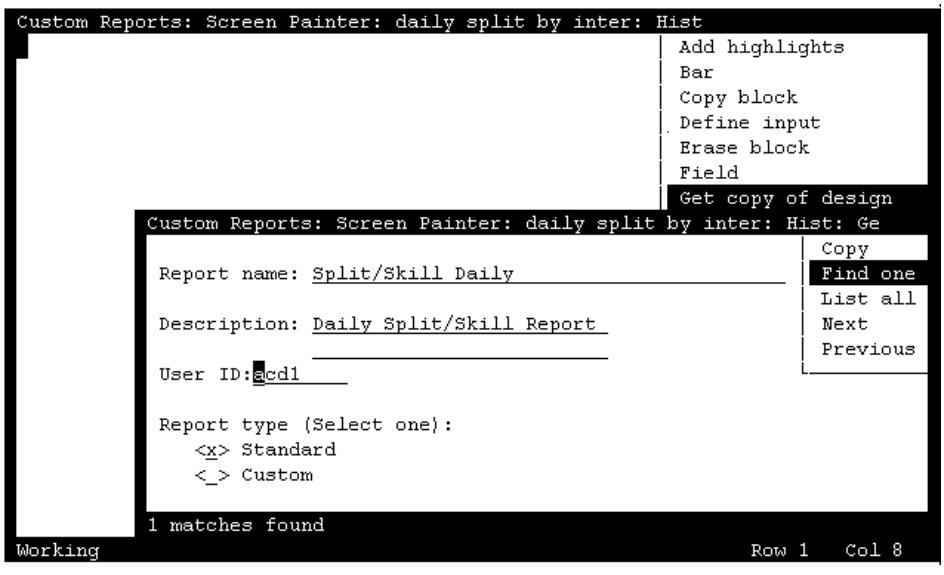


Figure 3-3: The Get Copy Window

Step 2: Enter a Report Name

In the `Report name` field, enter the name of the report you want to copy. You can copy standard report designs, global report designs, and private report designs. However, you **cannot** copy another user's private report designs unless you are a *CentreVu* CMS administrator.

If you do not know the report name, first select `List all`. For `List all`, you can leave the `Report name` blank, but you **must** complete the `Report type` field, described later. After you `List all`, you can go back and enter the report name.

Note

Because of the need to shorten names of standard reports for this window, the name you must enter in this window does not always match the exact names of standard reports.

Step 3: Enter a User ID

In the `User ID` field, enter a user ID only if one of the following is true:

- You are a *CentreVu* CMS administrator and you want to copy another user's private report.
- You want to list the custom reports of a specific user only.

Step 4: Select a Report Type

Enter **`x`** to select the `Report type` option (either `standard` or `custom`) for the type of report you want to copy. If the wrong option is selected, *CentreVu* CMS may not find or copy the report you actually want.

Step 5: Verify the Report

Select `Find one` to verify that the report is the one you want to copy. → *The user ID appears in the* `User ID` *field and, if it exists, the report description appears in the* `Description` *field.*

Step 6: Copy the Report

Select `Copy` to copy the report's design to the Screen Painter. → *The Get Copy window disappears and the report design appears on the Screen Painter, starting in the upper left corner.*

If you want to copy another report design, clear the upper-left area by deleting or moving any existing block of text and fields (see the "Editing a Report with Blocks" section in this chapter). Then repeat Steps 1 through 5.

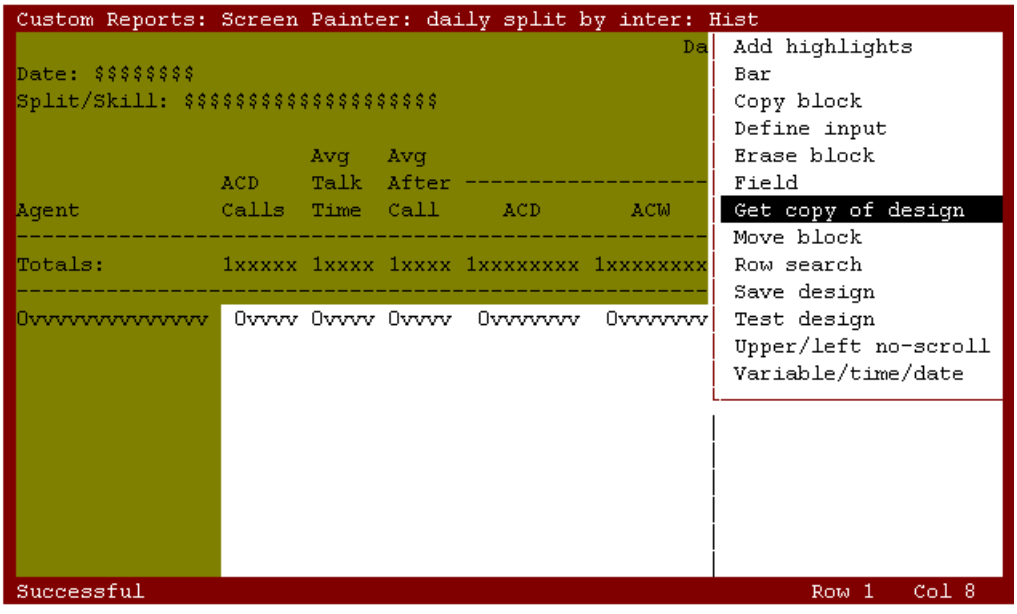


Figure 3-4: Case Study Sample — Copy of a Report Design (Using `Get copy`)

Note

If part of the report you are copying overlaps existing text or fields on the Screen Painter, *CentreVu* CMS does not copy that part of the report to the Screen Painter.

Editing a Report with Blocks

A **block** is a rectangular area on the Screen Painter that you define and use to quickly rearrange fields and text. You can erase blocks, copy blocks, or move blocks.

You define a block with the cursor by marking two opposite corners of a rectangle. When you define a block *CentreVu* CMS highlights the defined area. A block can contain a single character of text, a single field or bar, several words of text, several fields/bars, a combination of fields/bars and text, or a whole report design.

Note

If a block includes any part of a field/bar, then *CentreVu* CMS automatically includes the entire field in the block. You should therefore use care when deleting blocks to avoid unintentionally deleting a field/bar.

You can edit a report in blocks using the steps described in the following pages.

Erasing a Block

Note
Prior to completing a block erase, you can press Ctrl c to cancel the erasure.

1. On the Screen Painter, place the cursor in a position where you want one corner of the block to be, and select `Erase block`.
   → *The cursor returns to its original position, and the following message appears on the status line:* `Move cursor to define opposite corner of block, press RETURN.`
2. Move the cursor to a position where you want the opposite corner of the block to be. The block should include all fields and text you want to erase.
   → *The block is highlighted as you move the cursor.*

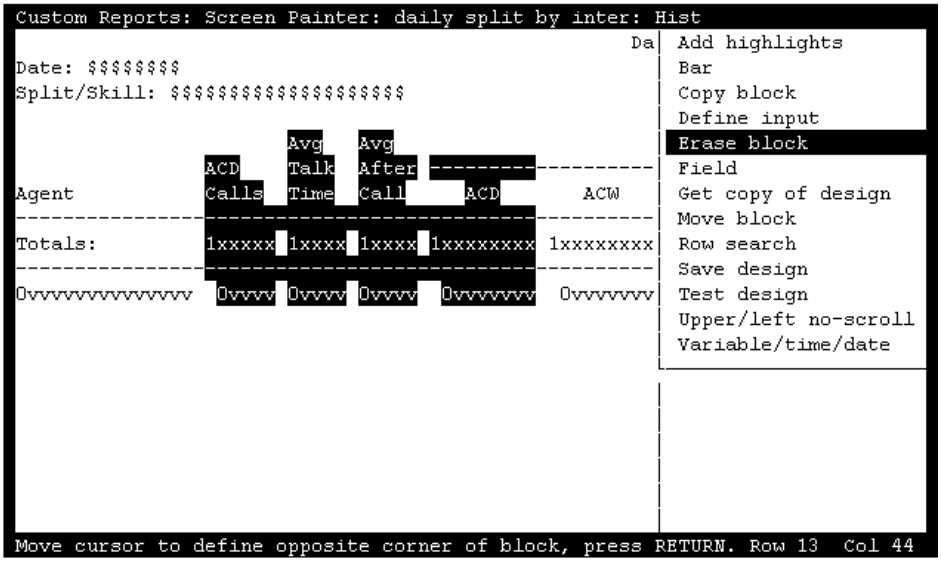


Figure 3-5: Sample Case Study — Defining a Block to Erase

Note
In our case study sample, we want to erase the middle section of the copied report design. Therefore, in Figure 3-5, we have scrolled horizontally to the middle section and defined the block we want to erase.

3. Press Return .
   → *An Acknowledgment window appears.*

4. Enter **y**, and press Return . → *All fields and text within the block disappear.*

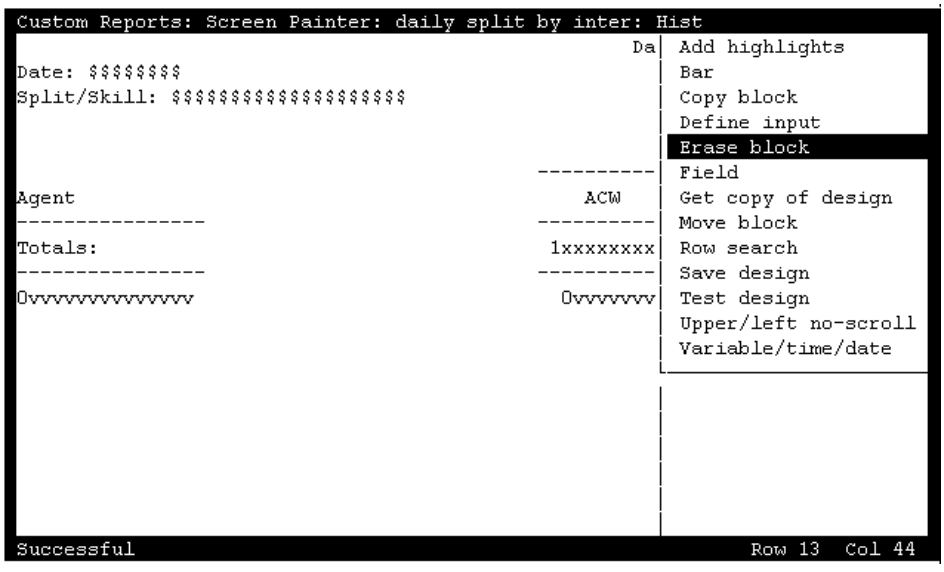


Figure 3-6: Sample Case Study — Erasing the Block

Moving a Block

Note

Prior to completing a block move, you can press Ctrl c to cancel the move.

1. On the Screen Painter, place the cursor in a position where you want one corner of the block to be, and select `Move block`. → *The cursor returns to its original position, and the following message appears on the status line:* `Move cursor to define opposite corner of block, press RETURN`.

2. Move the cursor to a position where you want the opposite corner of the block to be. The block should include all fields and text you want to move. → *The block you are defining becomes highlighted as you move the cursor.*

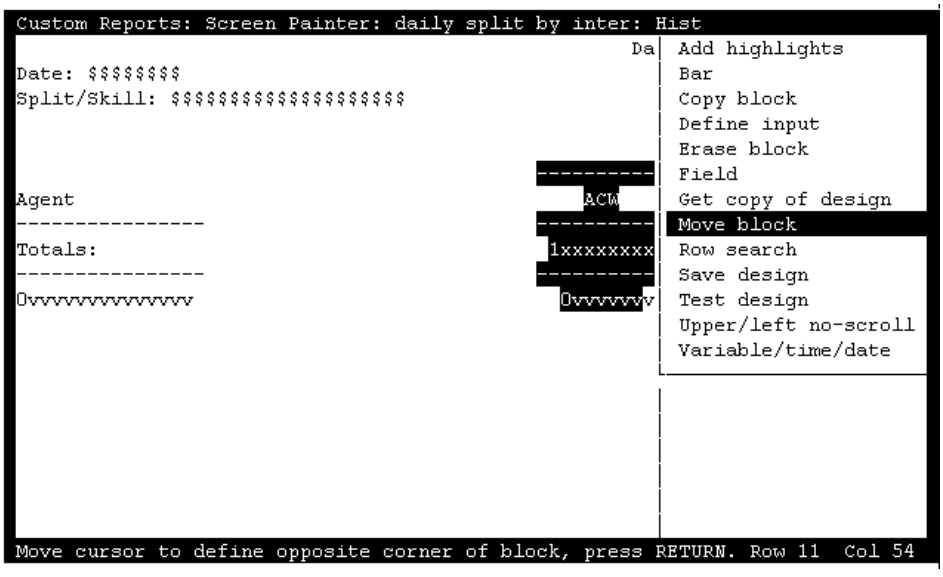


Figure 3-7: Sample Case Study — Defining a Block to Move

Note

In our case study, we want to keep the ACW time column, but we want to move it over to the left next to the other report items we want to keep.

3. Press **Return** . → *The following message appears in the status line:* `Move cursor to locate upper left corner of move, press RETURN.`

4. Move the cursor to a new position for the upper left corner of the block, and press **Return** . → *All fields and text within the block move to the new location.*

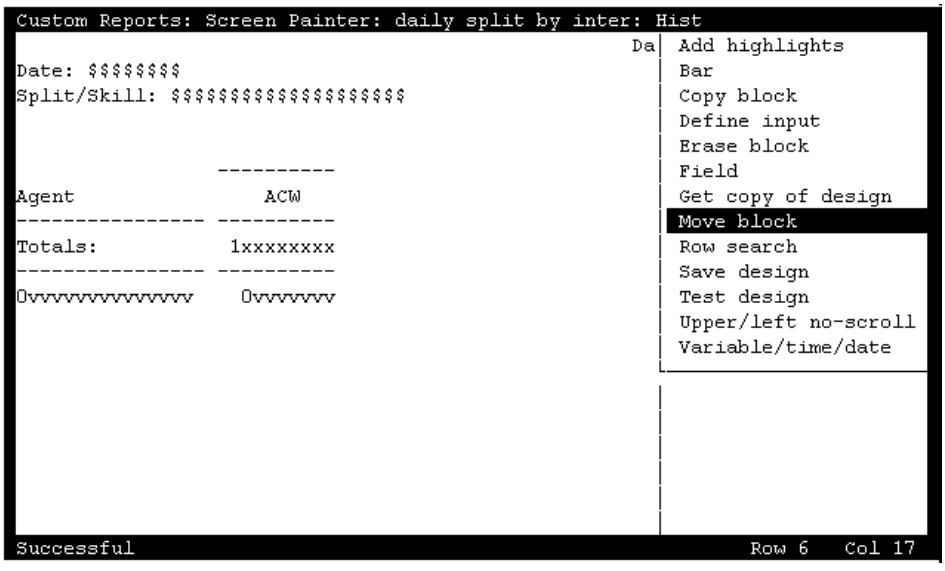


Figure 3-8: Case Study Sample — Completing a Block Move

When the block moves to its new location, the block's text or fields **cannot** overlap any other text or fields. If you do try to move a block to a location where it will overlap existing text or fields, an error message displays in the status line.

Copying a Block

Note
Prior to completing a block copy, you can press Ctrl c to cancel the copy.

1. On the Screen Painter, place the cursor in a position where you want one corner of the block to be, and select `Copy block`. → *The cursor returns to its original position, and the following message appears on the status line:* `Move cursor to define opposite corner of block, press RETURN.`
2. Move the cursor to a position where you want the opposite corner of the block to be. The block should include all fields and text you want to copy. → *The block you are defining becomes highlighted as you move the cursor.*
3. Press Return . → *The following message appears in the status line:* `Move cursor to locate upper left corner of copy, press Return.`
4. Move the cursor to a new position for the upper left corner of the block, and press Return . → *All fields and text within the block are copied to the new location.*

When you copy a block, the block's text or fields **cannot** overlap any other text or fields. If you do try to copy a block to a location where it will overlap existing text or fields, an error message displays in the status line.

Entering Report Text

You can enter text on the Screen Painter to label your fields, enter a report title, or include special instructions for the report. You should normally enter text to label each data field so that when you run the report you will know what data the field is showing.

You should enter text **before** defining the data fields for two reasons:

- The text provides a skeleton layout to help you position your data fields.
- The text helps you remember what each data field represents. If you have to change fields, this text saves you time.

To enter text, simply position the cursor where you want the text and type the characters. You can use the space bar to create spaces and to delete text. You can also simply overtype existing text with new text.

Note

You cannot overtype a data field.

To delete large portions of text, use the `Erase block` action list option (see the "Erasing a Block" section in this chapter).

Saving Your Work

If you want to save your work up to this point, select the `Save design` action and press the **Return** key.

If you press **Exit** before saving, then the following message appears:

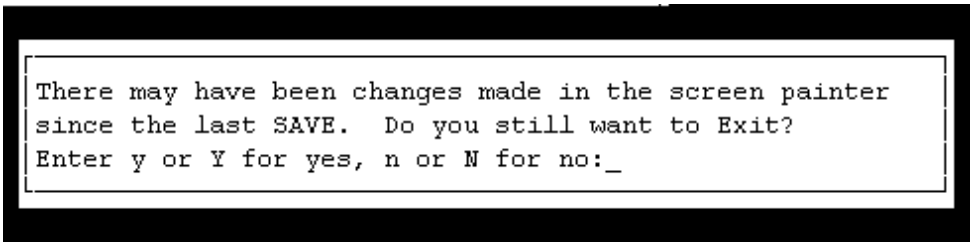


Figure 3-9: Save Report Message

If you enter **n** (or **N**), you are returned to the Screen Painter without saving any changes. If you enter **y** (or **Y**), you are returned to the Report Select window.

4 Defining the Data for a Custom Report

Defining Fields for the Report Input Window

To run a standard *CentreVu*™ Call Management System (*CentreVu* CMS) report, you first access a Report Input window (see Figure 4-1). The Report Input window gives you control over what data (which splits/skills, trunks, dates, intrahour intervals, and so on) are included in the report.

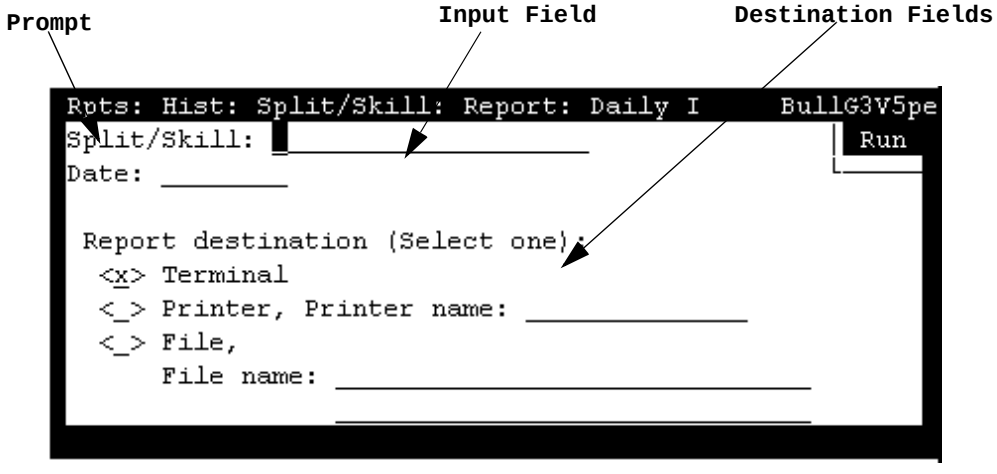


Figure 4-1: Sample Report Input Window

To define report input fields, use the Define Input window (Figure 4-2). You must complete a Define Input window for each input field. When you are done, you will have created a Report Input window. When you or another user prepares to run your custom report, this input window appears with the field prompt(s) and the input field(s) you defined (like those shown in Figure 4-1).

You define fields for the Report Input window using the steps described in the following pages.

Step 1: Access the Define Input Window

On the Screen Painter, select the `Define input` action list option. → *The Define Input window appears.*

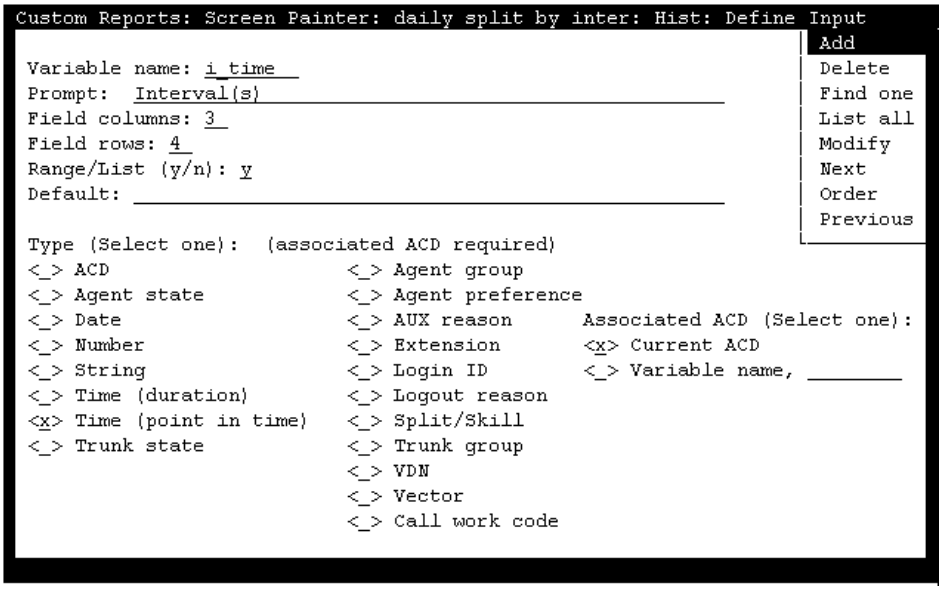


Figure 4-2: The Define Input Window (with Sample Inputs)

Step 2: Define the Variable Name

Enter a name of up to eight alphanumeric characters in the `Variable name` field. You must use this exact variable name again in the Row Search window when you define your row search conditions (see the "Defining the Rows of Data for a Report" section in this chapter). The variable name links the report input field to the row search conditions. This link enables *CentreVu* CMS to use the value(s) a user enters when running the report to search the database for appropriate report data. See Figure 4-3.

If you copy a report (via Get Copy), then that report's input fields and row search conditions are also copied. Before entering any variable names, you may first want to `List all` to see the report input fields that are already defined and the variable names they use. If you then add, delete, or change a variable name for a report input field, then you must also add, delete, or change that variable name in the row search conditions.

You cannot `List all` or `Find one` in the Define Input window by searching on multiple fields. For a `Find one` search, *CentreVu* CMS uses only the entry in the `Variable name` field. *CentreVu* CMS ignores the other fields. For a `List all` search, *CentreVu* CMS ignores all fields.

Figure 4-3 shows an example of how the report input fields are linked to a report's search conditions.

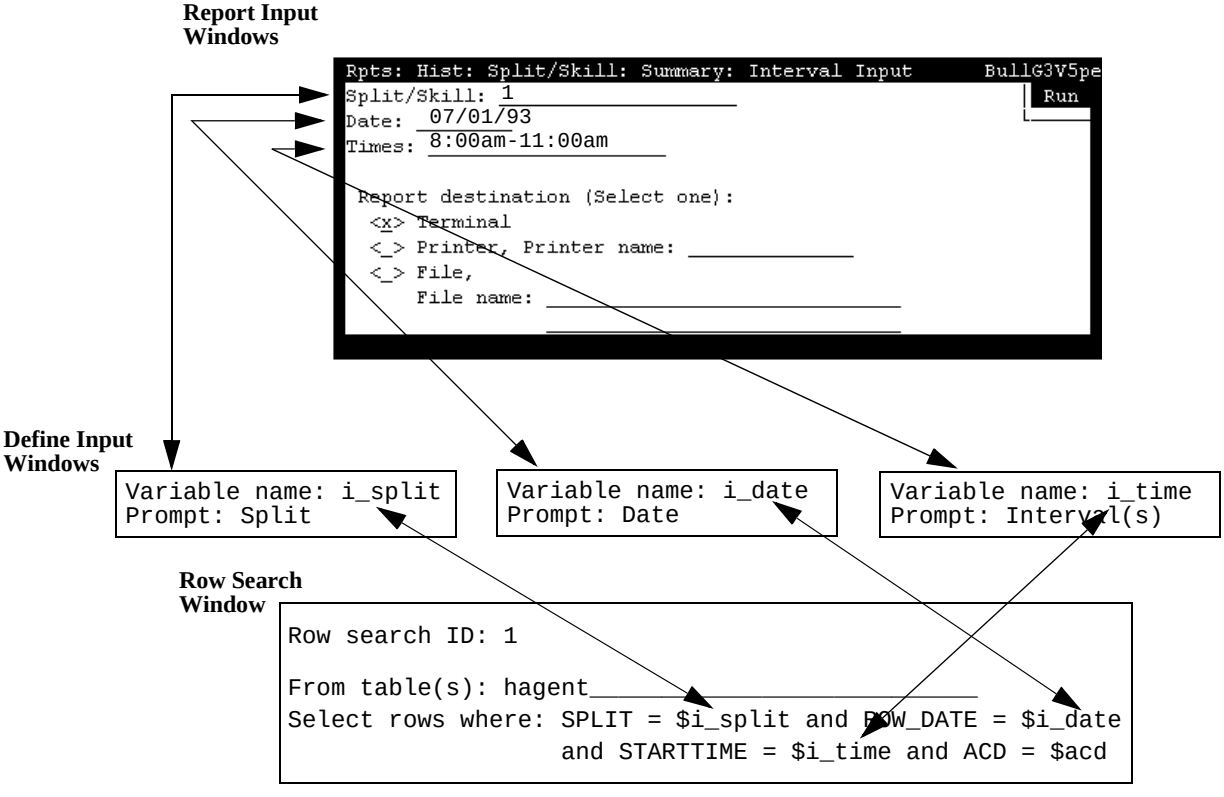


Figure 4-3: How a Variable Links the Report Input and Row Search

In the example, the user enters a split number of 1, a date of 07/01/93, and intrahour intervals from 8:00 a.m. to 11:00 a.m. *CentreVu* CMS identifies 1 as the value for variable name **`i_split`**, 07/01/93 as the value for variable name **`i_date`**, and 8:00 a.m. to 11:00 a.m. as the values for variable name **`i_time`**. *CentreVu* CMS then searches the specified database items (**`SPLIT`**, **`ROW_DATE`**, and **`STARTTIME`**) in the Intrahour Agent (**`hagent`**) table for rows that have those values. Finally, *CentreVu* CMS extracts data from those rows and displays the data in the report's fields.

Step 3: Enter a Field Prompt

In the `Prompt` field, enter a name of up to 50 characters to appear next to the input field on the Report Input window. Because you can use blanks, your prompt can be more than one word.

This name should describe the information (what split/skill, what date, what time, etc.) a user must enter in the field when ordering the report. For example, if you want the user to enter a date, “Date” would be an appropriate prompt. However, if the user can enter more than one date in the field, “Date(s)” would be more appropriate.

Step 4: Define the Number of Field Columns and Rows

The size of the input field is determined by the *product* of the numbers entered in the `Field columns` and the `Field rows` fields on the Report Input window. The maximum *product* of the two numbers you enter here cannot exceed 50. If the user will order the report with names (for example, split names) instead of numbers, make sure the field is large enough for the user to enter the complete name (up to 20 characters). If the user can enter a range of values, make sure the field is large enough for any range the user might possibly enter.

Step 5: Define the Field as Single Value or Range/List

Enter **`y`** in the `Range/List` field to allow the user to enter a range or list of values in the input field. If you enter **`n`**, the user will be able to enter only one value in the input field when ordering the report. For example, if you define a “Date” input field so that the report will show data for only one day, type **`n`**. If you define a “Times” or “Intervals” input field so the report can show data for multiple intrahour intervals in a day, type **`y`**.

Note

If you specify a range/list for a variable name in the Define Input window, then in the Row Search window you **must** use the equals (=) sign in the “where” clause for that variable name. See the “Defining the Rows of Data for a Report” section in this chapter.

Step 6: Provide a Default Value for the Field

Enter a value in the `Default` field of up to 50 characters. This value appears in the input field when the user first accesses the Report Input window. The user can then choose to overtype this value with another value or order the report with this value.

You can also leave the `Default` field blank. In this case, the input field will be blank when the user accesses the Report Input window.

Step 7: Select a Field Type

Enter an **`x`** to select an item in the `Type` list. The field type tells *CentreVu* CMS what kind of values it should expect the user to enter.

Field types are listed in Table 4-1.

If *CentreVu* CMS knows what to expect, *CentreVu* CMS can do the following:

- Check that the user's entries are valid system values and are values *CentreVu* CMS can use to search the database tables.
- Check that the user has permissions to run a report for the entries.
- Allow the user to enter names defined in the Dictionary subsystem.

For example, if you define an input field for the user to select a split/skill for the report by assigning the "Split/Skill" field type to the input field, the following would be possible:

- If the user enters a number, *CentreVu* CMS can check, **before searching the database**, to see that the number is within system limits (for example, 1 to 60 for System 85/G2).
- The user can enter the name of the split/skill as defined in the Dictionary subsystem. *CentreVu* CMS can check the `Dictionary: Split/Skill` list to find the split/skill number associated with the name.
- *CentreVu* CMS can check to see that the user has read permission for the split/skill.

Similarly, if your variable's input field requires a date, then by specifying a "Date" field type *CentreVu* CMS can check that the user does not enter a date like **`070193`** — which is not in a format that *CentreVu* CMS can use.

Table 4-1: Report Input Field Types

Type	Description
ACD	The user must enter an ACD number or name (as defined in the Dictionary subsystem).
Agent group	The user must enter an agent group name (as defined in the Dictionary subsystem).
Agent state	The user must enter an agent state name (standard or new name as defined in the Dictionary subsystem). Standard names are ACD, AUX, ACW, and so on.
Agent preference	The user must enter an agent call handling preference number or name.
AUX reason	The user must enter an AUX reason code name or number (as defined in the Dictionary subsystem).
Call Work Code	The user must enter a call work code name or number.
Date	The user must enter a date in mm/dd/yy format or as a relative number (for example, -7 for 7 days ago).
Extension	The user must enter an extension number of one to five digits (as administered for System 75/Generic 1/Generic 3) or three to five digits (as administered for System 85/Generic 2).
Login ID	The user must enter a login ID of one to nine digits (as administered for System 75/Generic 1/Generic 3) or four digits (as administered for System 85/Generic 2).
Logout reason	The user must enter a Logout reason code number or name (as defined in the Dictionary subsystem).
Number	The user enters a number, which may include digits to the right of the decimal point. This type applies if your variable field asks for specific values about ACD performance (for example, number of ACD calls or percent within service level).
Split/Skill	The user must enter a split/skill number or name.
String	The user enters a character string. Select this type only if one of the following is true: 1. Your variable field is linked to a custom database item that you identify in *INFORMIX** as a CHAR column. 2. Your variable field is linked to a standard database item that is a CHAR column, **AND** you want to allow the user to do pattern searching when running the report. See the following section, "Input Fields That Allow Pattern Matching".
Time (duration)	The user enters a number, including decimals, of seconds. This type only applies if your variable field asks for specific values regarding ACD performance (for example, time in AUX work, average speed of answer, or average talk time).
Time (point in time)	The user enters a specific time of day in hh:mm format, either as 24-hour time or with `am` or `pm` appended.
Trunk group	The user must enter the number or name of a trunk group.

Table 4-1: Report Input Field Types — *Continued*

Type	Description
Trunk state	The user must enter a trunk state name (standard name or new name as defined in the Dictionary subsystem). Standard names are `IDLE, SEIZED, QUEUED,` and so on.
VDN	The user must enter a Vector Directory Number of one to five digits (as administered for System 75/Generic 1/Generic 3) or three to five digits (as administered for System 85/Generic 2).
Vector	The user must enter a vector number or name.

* INFORMIX is a registered trademark of Informix Software, Inc.

Note

If you select a `String` or `Number` field type, *CentreVu* CMS does **not** accept any names (for VDNs, Splits/Skills, Login IDs, and so on) defined in the Dictionary subsystem. Also, *CentreVu* CMS does **not** check permissions or system limits.

The `Number` type allows the user to enter any number. The `String` type allows the user to enter any number, letter of the alphabet, or keyboard symbol in any format.

Note

The field type **does not** determine what database item(s) the variable represents. The database item(s) linked to the variable is specified in the Row Search window.

Input Fields That Allow Pattern Matching

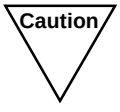
CentreVu CMS can search for values in certain database items according to wildcard search patterns. As a result, you can create a custom report that allows report inputs based on character strings, plus either ***** (matches 0 or more occurrences of any character) or **?** (matches on any single character). *CentreVu* CMS then includes data for all items that match the character strings the user entered.

The standard database items that allow this type of searching are:

- VDN (the value is a VDN number)
- EXTENSION (the value is an extension number)
- LOGID (the value is an agent login ID)
- EQLOC (the value is a 9-digit trunk location number)
- CWC (the value is a call work code)
- ROW_DATE (the value is a date)

In addition, any custom database items that you define as CHAR columns in *INFORMIX* also allow this type of searching.

As an example of matching with *****, if an input field were a `String` type and were associated with the LOGID database item, the user could enter **`1*`**, and *CentreVu* CMS would include data for all agents with login IDs that start with 1 (1, 10, 1238, 190, and so on, depending on the switch's administered login length). As an example of matching with **?**, if an input field were a `String` type and were associated with the VDN database item, the user could enter **`21?0`**, and *CentreVu* CMS would include data for all VDNs that start with 21, end with 0, and have any single character appearing between the 21 and the 0 (2100, 2110, 2120, 2130, and so on).



If you select `String` for an input field, *CentreVu* CMS does not check a user's inputs in that field for appropriate read permissions or valid switch parameters. If you want *CentreVu* CMS to check permissions for a VDN input field, you must select the **`VDN`** field type. If you want *CentreVu* CMS to check switch parameters for a VDN, login ID, extension, or call work code input field, you must select that field type, **not** `String`. In addition, if you select `String` for a field, the user will not be able to enter Dictionary names. So, again, if you want to let the user enter VDN, login ID, or call work code names to run a report, you must select that specific field type, **not** `String`.

Step 8: Associate an ACD with the Variable Field

Enter an **`x`** in the `Associated ACD` list to associate the variable field with either the current ACD or a user-selected ACD. You **must** select an associated ACD if the input field you are defining is a type listed under the heading **`(`**`associated ACD required`**`)`**. These field types require an associated ACD because they are administered for each ACD. For other field types not administered per ACD, *CentreVu* CMS ignores any selection of an associated ACD.

Select `Current ACD` if either of the following conditions is true:

- You have only one ACD.
- You always want the report to show data for the user's current ACD.

Select `Variable name` if you want to let the user select the ACD. If you select `Variable name`, you must also enter a name of up to eight characters in the field next to the `Variable name` list item.

Using a variable name for the associated ACD is most useful when you are creating a multi-ACD report.

Note

Select `Variable name` only if the user(s) who will run the report has read permission for more than one ACD. (You can create such a report without knowing if the user has read permission for more than one ACD. If you do and the user does not have read permission, the user will not be able to view the report.)

Before you can select `Variable name` for an input field's associated ACD, you must define a separate report input field for the ACD number/name. This field must have:

- An assigned field type of **`ACD`**
- The same variable name you are assigning to the associated ACD.

Step 9: Save a Variable Input Field

Select the `Add` action list option to save the definition of the variable input field.

Note

`Add` saves newly defined input fields. If you have previously saved an input field definition and you are changing it, then you must use `Modify` instead.

To define more input fields, press Ctrl Z to clear the Define Input window, and repeat steps 2 through 9.

Note

If you are defining a real-time report, you **cannot** and **do not need to** define an "Refresh Rate in Seconds" input field. *CentreVu* CMS puts this field in the Report Input window automatically.

If you are defining a historical report, you **cannot** and **do not need to** define fields for "Report Destination." *CentreVu* CMS puts this field in the Report Input window automatically.

Defining the Order in Which Input Fields Appear

On the Report Input window, *CentreVu* CMS displays your input fields in the same order you defined them in. However, you have the option of changing this order. To change the order in which the input fields appear, do the following steps.

1. After you have defined (**and saved**) all of your input fields, select the **`Order`** action list option.

   → *The Define Input: Order window appears. The window displays the prompts for each input field you have defined.*

Figure 4-4: The Define Input: Order Window

2. Enter **`1`** next to the prompt you want to appear first on the Report Input window. Enter **`2`** for the second prompt, **3** for the third, and so on.

3. Select the `Modify` action list option to save the display order.

   → *The Define Input: Order window disappears, and the cursor returns to its previous position on the Define Input window.*

Changing Report Input Fields

1. On the Screen Painter, select the `Define input` action list option. → *The Define Input window appears.*
2. Enter the desired name in the `Variable name` field, and select `Find one`. → *The defined characteristics of the variable name appear.*

   Before entering any variable names, you may first want to do a `List all` to see what report input fields you have already defined. If you copied an existing report design, report input fields defined for that report are also copied and available in the Define Input window.

   Note
   You cannot do a `List all` or a `Find one` search on multiple fields in the Define Input window. For a `Find one` search, you can only have an entry in the `Variable name` field. The other fields are ignored. For a `List all` search, all fields are ignored.

3. Change data in any field(s) except `Variable name`, and select the `Modify` action list option. → `Successful` *appears in the window's status line to indicate that the input field definition has been changed.*

Note
If you wish to change a variable name, you must delete the variable and then add a new one.

Defining Report Fields

A report consists of fields of data. Defining those fields is the central task of creating a custom report.

To define a field, you must specify the following items:

- The **specific data** that goes in the field.
- The **position** of the field
- The **length** of the field
- The **format** of the field's data

You do all of these tasks in a predefined sequence described in the following pages.

What the x's, v's, and h's Mean in a Field

If you copy the design of an existing report, the field(s) on the Screen Painter appears filled with x's, v's, or h's.

- x's indicate that the field is *discrete*. That is, the field appears as a single field in the report because, based on the row search conditions assigned to the field, *CentreVu* CMS finds only one value.
- v's indicate that the field is repeated *vertically* in the report. That is, the field appears as a column of fields because, based on the row search conditions assigned to the field, *CentreVu* CMS finds multiple values.
- h's indicate that the field is repeated *horizontally* in the report. That is, the field appears as a row of fields because, based on the row search conditions assigned to the field, *CentreVu* CMS finds multiple values.

See the "Assign a Row Search ID to Report Field(s) and Bar(s)" section in this chapter for illustrations of discrete and repeated fields.

Step 1: Define the Position and Length of a Field

1a. On the Screen Painter, position the cursor where you want a field to begin, and select `Field` from the action list.

→ *The cursor returns to its original position, and the following message appears*: `Move cursor to define opposite corner of field and press RETURN.`

1b. Move the cursor using the arrow keys to define a field length, and press **Return** . Be sure the field is long enough to contain the data. If the field is too short for a value, the report shows asterisks (*) in the field or, if the data is a word, cuts letters off.

→ The field appears as a question mark (**?**) followed by x's. The question mark indicates that you have not yet assigned a Row Search ID to the field. The Field window also appears.

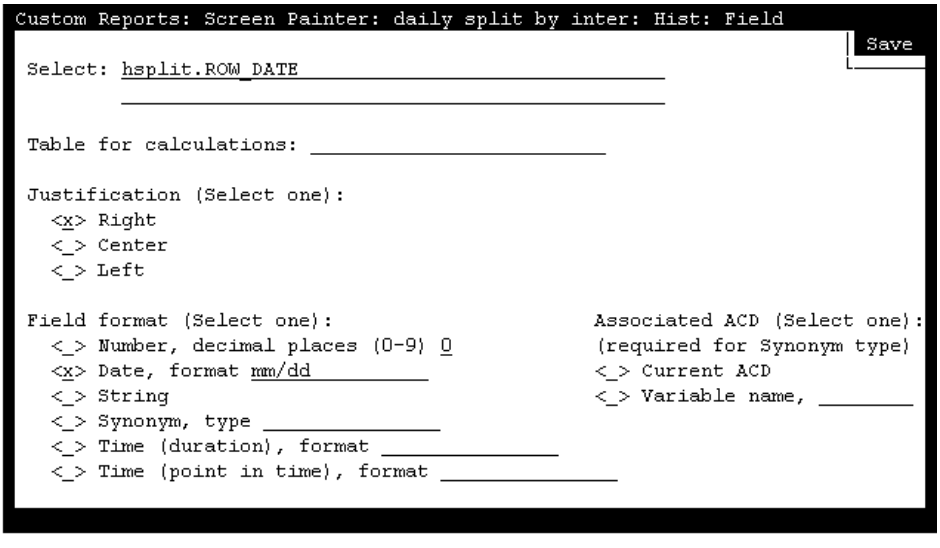


Figure 4-5: The Field Window

Step 2: Define the Field's Data Expression

In the `Select` field (shown in Figure 4-5), enter a data expression to tell *CentreVu* CMS two things:

- What table column(s) should supply data to the field.
- How to manipulate that data.

CentreVu CMS picks out values from a table with **both** row and column identifiers. *CentreVu* CMS identifies rows of data according to the user's inputs and the row search conditions you define (see the "Defining the Rows of Data for a Report" section in this chapter). *CentreVu* CMS identifies columns according to the data expression you define here.

You can enter the following types of data expressions:

Database Items

A database item is the name of a column of data in a table, either standard or custom. When you enter a database item, you must always add the name of a table and a period (.) as a prefix. The format is as follows:

```
<table name>.<database item>
```

Examples:

```
dsplit.ACDCALLS
hagent.STARTTIME
ctkgrp.NUMINUSE
```

Standard database items

Standard database items are listed in the Dictionary subsystem as having all upper-case letters (as in the preceding examples).

A standard database item can store:

- Identifiers (for example, **SPLIT**, **VDN**, **LOGID**, and so on)
- Timed data (for example, **ACDTIME**, **ABANTIME**, **AUXOUTIME**, and so on)
- Event counts (for example, **ACDCALLS**, **INTERFLOWCALLS**, **ABNCALLS**, and so on)
- For real-time and agent trace reports, current state data (for example, **WORKMODE**, **DURATION**, **NUMINUSE**, and so on).

See Appendix A for a description of database tables and items including the exceptions, forecast, and login/logout tables. See Chapter 6, "Advanced Report Design", for more information about custom reports that include exceptions and forecast data.

Note

Standard database items are often shared by more than one table. For example, **ABNCALLS** can identify a column in the Current Interval Split, Daily Split, or Intrahour Agent tables (or many other tables).

CentreVu CMS can determine the exact database item only when it is identified with a table.

Custom database items

You must enter a custom database item, with the custom table name as a prefix, exactly as you defined it in the Dictionary subsystem. The data identified by a custom database item depends entirely on the data you entered for the item in the custom table (see Chapter 6, "Advanced Report Design").

Constants

A constant is the name of a fixed numerical value (whole number or decimal) that you define in the Dictionary subsystem. Constant names can be up to 20 characters long. A constant could represent a per-minute usage rate for trunks, a daily or hourly wage rate, or a service objective (like number of abandons, number of ACD calls, or percent within service level). A constant could also represent an average for the estimated dollar loss of an abandoned call, which could then be used to calculate daily loss of revenue due to abandoned calls. No standard constants exist in *CentreVu* CMS when it is first installed. Therefore, you must define every constant you want to use.

Using constants makes sense only if you have a fixed value that you want to use under one or both of the following conditions:

- The constant is a value that you will use in a number of different custom reports (for example, an average wage rate).
- You would not be able to remember the numerical value, but could remember a name assigned to the value (for example, for the $9.00 hourly wage rate for an agent called Smith, you could have a constant called **Smithwage**.)

Note

The *CentreVu* CMS real-time database only allows whole numbers in queries. If you need a value to be a decimal (for example, 9.5), use whole numbers and division to arrive at the correct number (so, in order to have 9.5 in a query, you would use 19/2 as the query entry).

Calculations

A calculation is a combination of database items and arithmetic operators. You can also include constants in a calculation. The arithmetic operators are:

+	add
-	subtract
*	multiply
/	divide
()	perform first

Some examples of calculations are:

```
dsplit.ACDCALLS/dsplit.ACDTIME

hagent.AUXOUTTIME+hagent.ACWOUTTIME

100*((cagent.I_ACDTIME+cagent.I_ACWTIME)/
cagent.I_STAFFTIME)
```

Arithmetic operations are generally performed in order from left to right. However, multiplication and division operations are performed before addition and subtraction operations, unless the addition or subtraction operations are enclosed in parentheses. Operations in parentheses are always performed first. If more than one set of parentheses is used, the operation in the set farthest to the left is performed first. If one set of parentheses is inside of another set, the operation of the inner set is performed first.

Calculation names

A calculation name is a name, as defined in the Dictionary subsystem, that can substitute for the actual calculation. The calculation name can be a standard name (used in standard reports) or a name you define. You cannot append a table name to a calculation name. Therefore, you must specify a table name in the `Table for calculations` field.

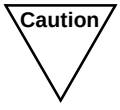
Note

CentreVu CMS differentiates between uppercase and lowercase letters in calculation names. Therefore, be sure you enter the desired calculation name *exactly* as it appears in the Dictionary subsystem.

A calculation name normally reflects the purpose of the calculation. As a result, entering a name is an easier, more meaningful way to define data for a report field. More importantly, if you use a calculation name in many custom reports and later decide to change the calculation, you can simply make your changes once in the Dictionary subsystem. *CentreVu* CMS then applies those changes to every report that uses the calculation name.

For example, if you use the standard calculation name <PERCENT_ACD_TIME>, which represents the calculation

100*((I_ACDTIME+I_ACWTIME)/I_STAFFTIME), but do not want to include the ACWTIME in the calculation, then you can change the calculation in the Dictionary subsystem so that <PERCENT_ACD_TIME> represents 100*(I_ACDTIME/I_STAFFTIME). Then, any report that uses the calculation name PERCENT_ACD_TIME reflects the new calculation.



If you change the calculation for a standard calculation name, the change affects any standard report, as well as any custom report, that uses that calculation name.

At times, using calculation names helps save space in the `Select` field so you can create more complex calculations. For example, if you wanted to know the average time agents spent on all extension calls (both incoming and outgoing), you might have to add the following calculation:

`(csplit.ACWINTIME + csplit.AUXINTIME + csplit.ACWOUTTIME + csplit.AUXOUTTIME) / (csplit.ACWINCALLS + csplit.AUXINCALLS + csplit.ACWOUTCALLS + csplit.AUXOUTCALLS)`

Unfortunately, the `Select` field is not long enough to enter the complete calculation. However, you could define in the Dictionary subsystem two separate calculation names for each half of the calculation.

For:

`(csplit.ACWINTIME + csplit.AUXINTIME + csplit.ACWOUTTIME + csplit.AUXOUTTIME)`,

you could enter, in the Dictionary subsystem, the calculation name:

`TIME_ON_NON-ACD_SUM with the assigned calculation (ACWINTIME + AUXINTIME + ACWOUTTIME + AUXOUTTIME)`.

Likewise, for:

`csplit.ACWINCALLS + csplit.AUXINCALLS + csplit.ACWOUTCALLS + csplit.AUXOUTCALLS)`,

you could enter the calculation name:

`NON-ACD_CALLS_SUM with the assigned calculation (ACWINCALLS + AUXINCALLS + ACWOUTCALLS + AUXOUTCALLS)`.

As a result, you can enter the following calculation in the **`Select`** field:

`TIME_ON_NON-ACD_SUM/NON-ACD_CALLS_SUM`

Then, you can enter the table name **`csplit`** in the `Table for calculation` field.

Caution

You **should not** add table names to your custom calculations in the Dictionary subsystem. Doing so makes the assigned calculation name less flexible for use in custom reports. Also, if you append table names to the Dictionary calculation and then also assign a table name to the calculation name in the Field window, the report will fail.

Aggregate Functions

An aggregate function is a prefix attached to a database item, a calculation, parts of a calculation, or a calculation name. When you define an aggregate function, you must place the database item or calculation in parentheses [as in, **`max(dsplit.ACDTIME/dsplit.ACDCALLS)`**].

Note

In real-time reports, fields with aggregate functions **cannot** share a set of row search conditions with nonaggregate fields. And for historical reports, special considerations exist when you assign the same row search conditions to both aggregate functions and other types of data expressions. See "Repeating Aggregate Function Values" in Chapter 6 of this manual.

An aggregate function can be one of four types. Each type retrieves a different value from the data.

- **max**

  The **max** aggregate function retrieves the highest value for a calculation or database item over the time frame of the report.

  For example, if the Intrahour Split table contained data as shown in Figure 4-6 and you entered **`max(hsplit.ACDCALLS)`** for a field in a report, then ran the report for Split 1 for all intervals on 07/02/93, *CentreVu* CMS would find all rows shown in bold. However, *CentreVu* CMS would display only the value **418** (shown in the box), which is the maximum ACD calls in any single interval on 07/02/93. Likewise, if you entered **`max(ACDTIME/ACDCALLS)`** for the field, *CentreVu* CMS would display the value **101.53** (which is the highest average talk time in any single interval on 07/02/93).

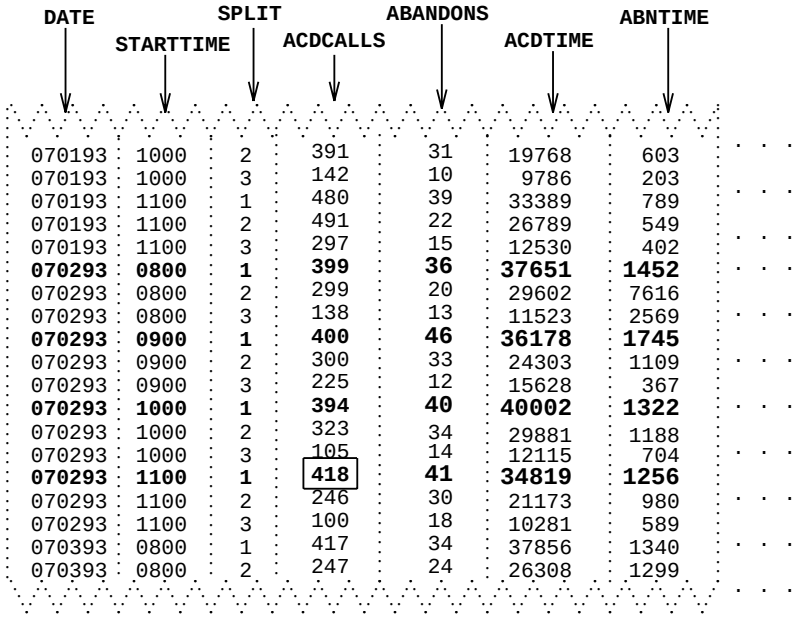


Figure 4-6: Sample Intrahour Split Table Data

- **min**

  The **min** aggregate function retrieves the lowest value for a calculation or database item over the time frame of the report.

  For example, using the previous example, if you entered **`min(hsplit.ACDCALLS)`** instead of **`max(hsplit.ACDCALLS)`** for the field and then ran the report for Split 1 for all intervals on 07/02/93, *CentreVu* CMS would display only the value **394** which is the smallest number of ACD calls in any single interval on 07/02/93.

- **sum**

  The **sum** aggregate function retrieves the sum of all values for a calculation or database item over the time frame of the report.

  For example, if the Intrahour Split table contained data as shown in Figure 4-6 and you enter **`sum(hsplit.ACDCALLS)`** for a field in a report, then for Split 1 and all intervals on 07/02/93, *CentreVu* CMS would take the values for **`hsplit.ACDCALLS`** and add them up to display only the value **1611**. Likewise, if you entered **`sum(hsplit.ABANDONED+hsplit.ACDCALLS)`** for the field, *CentreVu* CMS would display only the value **1774**, which is the total of all ACD calls and abandons for Split 1 on 07/02/93.

- **avg**

  The **avg** aggregate function retrieves the average of all values found over the time frame of the report. Using the sample table in Figure 4-6, if you enter **avg(ACDCALLS)** for a field and run the report for Split 1 for all intervals on 07/02/95, the value **402.75**, which is the average of **399**, **400**, **394**, and **418**, is displayed.

count(*)

The **count(*)** expression tells *CentreVu* CMS to count the number of rows in a table that match certain row search conditions (as defined in the Row Search window). For example, if you want the number of agents in a split that had more than five extension-out calls, the Row Search window would have a row search statement like:

```
Select rows where: SPLIT = 1 and
                   (ACWOUTCALLS+AUXOUTCALLS) > 5
```

This statement means, “Find rows of data where the SPLIT value is **1** and total extension-out calls, for both ACW and AUX states, is greater than 5." Then, by entering **count(*)** in the Select: field, the report field would count the number of rows that match, and display the number in your report.

Do not append a table name to the beginning of **count(*)**.

count(*) is always a number of matching rows and makes sense only if you want to track some specific measure of performance by ACD elements (for example, the number of agents currently logged into a split, the number of trunks that were occupied for more than 80% of the time, the number of VDNs that had over 30 abandoned calls in an intrahour interval).

count(*) allows you to create fields that act as exception counts.

Data from More Than One Table

A calculation can merge data from more than one table in a report field.

For example, you may want the percentage of a split's ACD calls that an agent handled in a day. Thus, you can enter a calculation that merges data from the Daily Agent and Daily Split tables, as in the following example.

```
dagent.ACDCALLS/dsplit.ACDCALLS
```

When you merge data from two tables, you must define your row search conditions in a special way. See “Selecting Rows from More Than One Table” in Chapter 6 of this manual.

Note

You cannot use calculation names for a field in which you merge data from two tables, and you cannot merge data in real-time reports.

Step 3: Define the Table(s) for Calculation Names

Enter a table name in the `Table for calculations` field **only** if you entered a calculation name in the `Select` field. The table name tells *CentreVu* CMS in what table to look for the database items in the calculation.

Note

The `Table for calculations` field is necessary because you **cannot** append a table name to a calculation name in the `Select` field.

For example, look at the following entries:

```
Select: AVG_POS_STAFF
Table for calculation: hsplit
```

These entries mean “Take the calculation defined in the Dictionary subsystem for <AVG_POS_STAFF>, which is **`I_STAFFTIME/(INTERVAL*60)`**, and apply the `hsplit` table name to the database items.” In effect, the two fields make the calculation **`hsplit.I_STAFFTIME/(hsplit.INTERVAL*60)`**.

Step 4: Justify Data in the Field

Select, from the `Justification` list, the way you want *CentreVu* CMS to line up data when the data appear in the field. Normally, numerical data is right-justified so that the right hand side lines up in a column. Names are normally left-justified so that the first character of each name is lined up. However, you may choose any of the three options for any type of data.

Step 5: Define the Field Format

Enter an **x** to select an item in the Field Format list. You must also complete the field associated with the list item.

The format type and the format you specify in the associated field tell *CentreVu* CMS how to display the values it finds for the field. However, the format you select depends on the type of data *CentreVu* CMS will display. The format options are as follows:

Number Select **Number** if the field will display a number of events, an average, or a percentage. You must also specify a number of decimal places for the field. Enter **0** if you do not need decimal places displayed. If the field's expression were **ACDCALLS**, you would select **Number** and enter **0** in the field. However, if the field expression were to generate an average, such as the average staffed positions per interval, **I_STAFFTIME/(INTERVAL*60)**, you may want to include decimal places.

When you run the report, the decimal point and the decimal places will use up spaces in the field. For example, if the field contains six spaces and you specify three decimal places for the field, then data will appear with two characters to the left and three characters to the right of the decimal point (for example, **12.344**).

Date Select Date if the field expression is **ROW_DATE**. You must also specify a date format, with appropriate punctuation. You can select a single format or a combination of formats. The available formats are:

Table 4-2: Date Formats for Report Fields

mm	=	Numerical month (for example, **12** for December)
MMM	=	Month represented by three letters (for example, **APR** for April).
yy	=	Year as two digits (for example, **94**).
YYYY	=	Year as four digits (for example, **1994**).
dd	=	Numerical day of the month (for example, **31**).
jjj	=	Day of the year in the Julian calendar (for example, **151** for May 31).
www	=	Day of the week as three letters (for example, **THU**).

`String`

Select **`String`** for those database items whose data *CentreVu* CMS identifies as character strings, not numbers. (Each of these items is identified as being a CHAR column type in *INFORMIX* terminology.) Even though these items store numbers, *CentreVu* CMS searches for values as if the items stored nonnumeric symbols and alphabetic words, as well as numbers.

Database items for which you might select `String` are as follows:

- **`VDN`** (the value is a VDN number)
- **`EXTENSION`** (the value is an extension number)
- **`LOGID`** (the value is an agent login ID)
- **`EQLOC`** (the value is a 9-digit trunk location number)
- **`CWC`** (the value is a call work code)
- **`Custom database items`** with the **CHAR** column type.

String, as used here, **does not** have the same meaning as string-value database item, as defined for the Dictionary subsystem. For the purposes of custom report design, Dictionary names for string-value database items are referred to as **`Synonyms`**, and include names for agents, splits, VDNs, trunk groups, and vectors. In fact, for the **`VDN`** and **`LOGID`** database items, you may wish to select **`Synonym`**, not **`String`**, since you may have assigned names to VDNs and login IDs in the Dictionary subsystem.

`Synonym`

Select `Synonym` to display a name defined in the Dictionary subsystem, instead of the value stored in the database table(s). You must also enter the Dictionary name type.

See the “Dictionary” chapter in the *CentreVu* ™ *CMS R3V5 Administration* (585-215-820) document for a complete description of synonym types.

The type must correspond to the database item you enter in the `Select` field. The types you enter are as follows:

Table 4-3: Synonym Types for Report Fields

Report Field	Type	Database Item
ACD Entities	acd	ACD
	agname	LOGID
	tkgrp	TKGRP
	split	SPLIT[b]
	vdn[a]	VDN
	vector[a]	VECTOR
	aux_rsn	AUXREASON
	logout_rsn	LOGOUTREASON
	cwc	CWC
Agent States	workmode[c]	WORKMODE
	ag_orig[d]	ORIGIN
	ag_dir	DIRECTION
	ag_dest	DESTINATION
	ag_pref	PREFERENCE
Trunk States	tkstate	TKSTATE
	tk_pri	PRIORITY
	tk_qtype	QUETYPE
	tk_vpri[a]	PRIORITY
	tk_dir	DIRECTION
	all_busy	ALLINUSE
Split States	slvl_chg	SVCLEVELCHG
	per_chg	PERIODCHG

a. Available only with the *CentreVu* CMS Vectoring feature.
b. Contains skill or split values.
c. Applies to the WORKMODE database item in both the Agent and Agent Trace tables.
d. Available only with adjunct routing on a Generic 3.

Time (duration) Select `Time (duration)` if the field expression will display a length of time. You must also enter a time format with the appropriate punctuation.

Durations (for example, **ACDTIME**) are stored as a number of seconds. However, you can choose to display time as minutes and seconds, or even hours, minutes, and seconds.

The time formats available are:

Table 4-4: Time Formats for Report Fields

ss	Display time as the number of seconds. Enter as many **s**'s as needed to display the seconds. If the number of seconds can reach six digits, enter **ssssss**.
mm	Display time as the number of minutes. Enter as many **m**'s as needed to display the minutes. If the number of minutes can reach six digits, enter **mmmmmm**.
hh	Display time as the number of hours. Enter as many **h**'s as there are digits needed to display the hours. If the number of hours can reach three digits, enter **hhh**.
mm:ss	Display time as minutes and seconds. The minutes count is increased by one and the seconds count is reset to 00 when the seconds count reaches 60. Specify more than two digits for minutes if the minutes will exceed 99. For example, if you enter **mmmm:ss**, 2822:35 (2822 minutes and 35 seconds) might display in the report.
hh:mm	Display time as hours and minutes. The hours count is increased by one and the minutes count is reset to 00 when the count reaches 60. Specify more than two digits for hours if hours will exceed 99. For example, if you enter **hhh:mm**, up to 999:59 can display in the report.
hh:mm:ss	Display time as hours, minutes, and seconds. The minutes count is increased by one when the seconds count reaches 60. The hours count increases by one when the minutes count reaches 60. With this format, you can increase the digits for hours if necessary, but **not** the minutes digits.

Time (point in time)

Select `Time (point in time)` if the field expression will display a point in time (for example, **`10:34am`**). Enter a time format with the appropriate punctuation.

You can use one of the following formats:

Table 4-5: Point-In-Time Formats for Report Fields

`HH`	The hour only, in military time (24-hour clock). For example, 15 equals 3:00 p.m.
`hh`	The hour only, according to a 12-hour clock. For example, 3 could mean 3:00 a.m. or 3:00 p.m. For this reason, if you use hh, you should also add am (hham).
`mm`	The number of minutes after the hour only.
`ss`	The number of seconds into the minute.
`HH:mm:ss` or **`HH:mm`**	24-hour clock time, either to the second or to the minute.
`hh:mm:ssam` or **`hh:mmam`**	12-hour clock time, with AM or PM attached, either up to the second or up to the minute.

Step 6: Save the Field Definition

Select the `Save` action list option. → *The Field window disappears,* and `Successful` *appears in the Screen Painter's status line to indicate the field definition has been added.*

To define additional fields, repeat steps 1 through 6.

Note

You must assign a Row Search ID to the field before your field definition is truly complete. When you do, the question mark (?) will change to that ID number. See the "Defining the Rows of Data for a Report" section in this chapter.

Changing a Field Definition

1. On the Screen Painter, place the cursor on the field you want to change, and select `Field`. → *The cursor returns to the field and rests on the last space of the field. The following message appears on the Screen Painter status line:* `Move cursor to define opposite corner of field, press RETURN.`
2. If desired, move the cursor using the arrow keys to either make the field longer or shorter, and press **Return**. → *The Field window appears.*
3. Overtype any data in fields you want to change, and select `Save`. → The Field window disappears, `Successful` appears in the Screen Painter status line, and the cursor returns to the field you just changed.

Defining Bars in a Report

Note

You can define bars in a custom report **only if** you have purchased the *CentreVu* CMS Graphics feature. If you have not purchased the Graphics feature, the **`Bar`** action list option is **not** available.

You may want a report to display data as bar graphs instead of numbers. Defining a bar or bars in a report is similar to defining fields, except in the way you define the format. To define a bar's format, you must specify the following items:

- The position and length of the bar
- The direction of the bar — horizontal or vertical
- The thresholds that cause the bar to change color
- The scale of the bar.

What the X's, V's, and H's Mean in a Bar

If you copy the design of an existing bar graph, the bar(s) appear filled with X's, V's, or H's.

- X's indicate that the bar is *discrete*. The bar appears as a single bar in the report because, based on the row search conditions assigned to the bar, *CentreVu* CMS finds only one value.
- V's indicate that the bar is repeated *vertically* in the report. The bar appears as a series of bars, one over the other, because, based on the row search conditions assigned to the bar, *CentreVu* CMS finds multiple values.
- H's indicate that the bar is repeated *horizontally* in the report. The bar appears as a series of bars, side by side, because, based on the row search conditions assigned to the bar, *CentreVu* CMS finds multiple values.

See the "Assign a Row Search ID to Report Field(s) and Bar(s)" section in this chapter for illustrations of discrete and repeated bars.

Step 1: Define the Position and Length of a Bar

1a. On the Screen Painter, position the cursor where you want a bar to begin, and select the **Bar** action list option.

→ *The cursor returns to its original position, and the following message appears:*

```
Move cursor to define
opposite corner of bar
and press RETURN.
```

1b. Move the cursor using the arrow keys to define the length and width of the bar, and press **Return**.

→ *A question mark (?) appears in the upper left corner of the bar. The question mark indicates that you have not yet assigned a Row Search ID to the bar.*

The Bar window also appears.

```
Custom Reports: Screen Painter: daily split by inter: Hist: Bar
                                                        List inputs
Select: hsplit.ACDCALLS                                 Save
        ____________________________________________

Table for calculations: ____________________________

Bar direction (Select one):          Scale (Select one):
  <_> Horizontal                       <x> Scale with tick marks
  <x> Vertical                         <_> Scale without tick marks
                                       <_> No scale

Maximum graph value (Select one):    First threshold (Select one):
  <_> Fixed ______                     <x> None
  <x> Variable name i_maxqr            <_> Fixed ______
                                       <_> Variable name ________
Second threshold (Select one):
  <_> None                           Reverse threshold colors (y/n): n
  <_> Fixed ______
  <x> Variable name i_thresh
```

Figure 4-7: The Bar Window (with Sample Input)

Step 2: Define the Bar's Data Expression

Enter a data expression in the `Select` field to tell *CentreVu* CMS two things:

- What table column(s) should supply data to the bar
- How to manipulate that data.

The rules for this `Select` field are identical to those of the `Select` field for the Field window. However, the following types of database items do not make sense for bars:

- Identifiers (for example, **`SPLIT`**, **`VDN`**, **`LOGID`**, and so on)
- Current state data (for example, **`WORKMODE`**, or **`DURATION`**)
- Constants (unless they are part of a calculation).

Also, if a bar's expression is a database item that stores a number of seconds, the bar normally shows time as seconds. However, you can make the bar represent minutes by dividing the database item by 60.

In addition, to complete your definition of bar data, you must, as for field data, define row search conditions for the bar(s). This includes whether a bar you define will be repeated to display multiple bars for multiple values. See the "Defining the Rows of Data for a Report" section in this chapter.

Step 3: Define the Table(s) for Calculation Names

Enter a table name in the `Table for calculations` field **only** if you entered a calculation name in the `Select` field. The table name tells *CentreVu* CMS in what table to look for the database items in the calculation.

Note

The `Table for calculations` field is necessary when using calculation names because you **cannot** append a table name to a calculation name in the **`Select`** field.

Step 4: Define the Bar Direction

Enter an **`x`** in the `Bar direction` list to select either a horizontal or vertical format for the bar. `Vertical` means the bar's length is oriented up and down. `Horizontal` means the bar's length is oriented left and right.

Step 5: Define the Bar Scale

Enter an **x** to select an option in the Scale list.

- **Scale with tick marks**

  The scale appears in the report as a line, segmented by evenly spaced marks (see Figure 4-8). For a vertical scale, a tick mark appears for each vertical character space. For a horizontal scale, a tick mark appears at every fifth horizontal character space.

- **Scale without tick marks**

  The scale automatically appears in the report as a line parallel to the bar (see Figure 4-8). The line will be a fixed length equal to the maximum length of the bar. In this way, the end of the scale always represents the maximum graph value you define.

- **No scale**

  The bar does not have a scale (see Figure 4-8). You may want this option if the scale defined for another bar applies to the bar you are currently defining. For example, in Figure 4-8, the scale for the ACD Calls bar could apply to the Abandons bar.

  A scale defined for one bar appears in correct relation to other bars only if the following are true:

  — All bars are parallel, have the same maximum length, and have the same starting points.

  — All bars have the same maximum graph value.

  — All bars display the same units (number of calls, averages, percentages, number of seconds, and so on).

Note

Do not enter any text or fields in the column immediately to the left of a vertical bar. And, **do not** enter any text or fields in the row immediately above a horizontal bar.

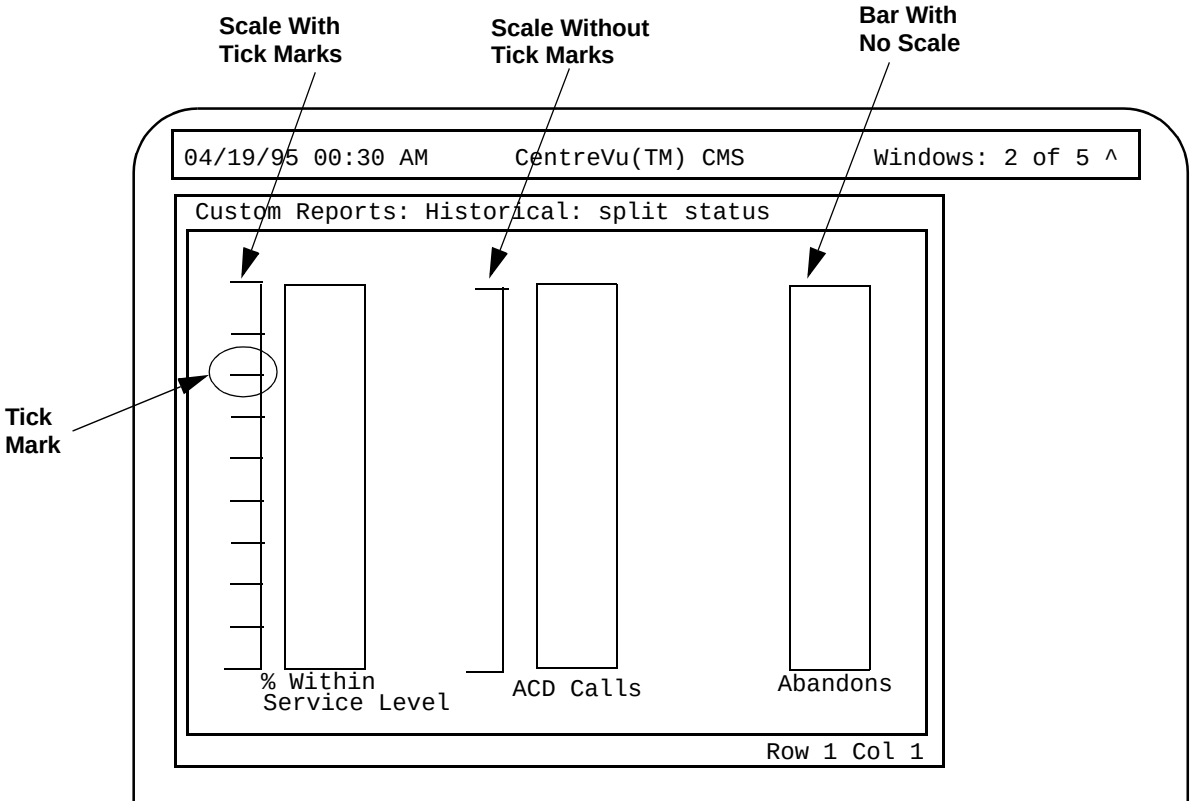


Figure 4-8: Bars — With and Without Scales

Step 6: Define the Maximum Graph Value

Enter an **x** in the Maximum Graph Value list to select either a fixed or variable maximum graph value. The maximum graph value is the value that the bar represents when the bar is at its maximum length or height (see Figure 4-9). The value must always be a whole number or decimal.

If you select Fixed, you must also enter, in the accompanying field, the value the bar should represent when the bar is at its maximum length or height.

If you select Variable name, you must also enter, in the accompanying field, a variable name that references a report input field. Thus, selecting Variable name allows the user to enter a maximum graph value for the bar when ordering the report. The variable name you enter must be identical to a variable name defined in the Define Input window.

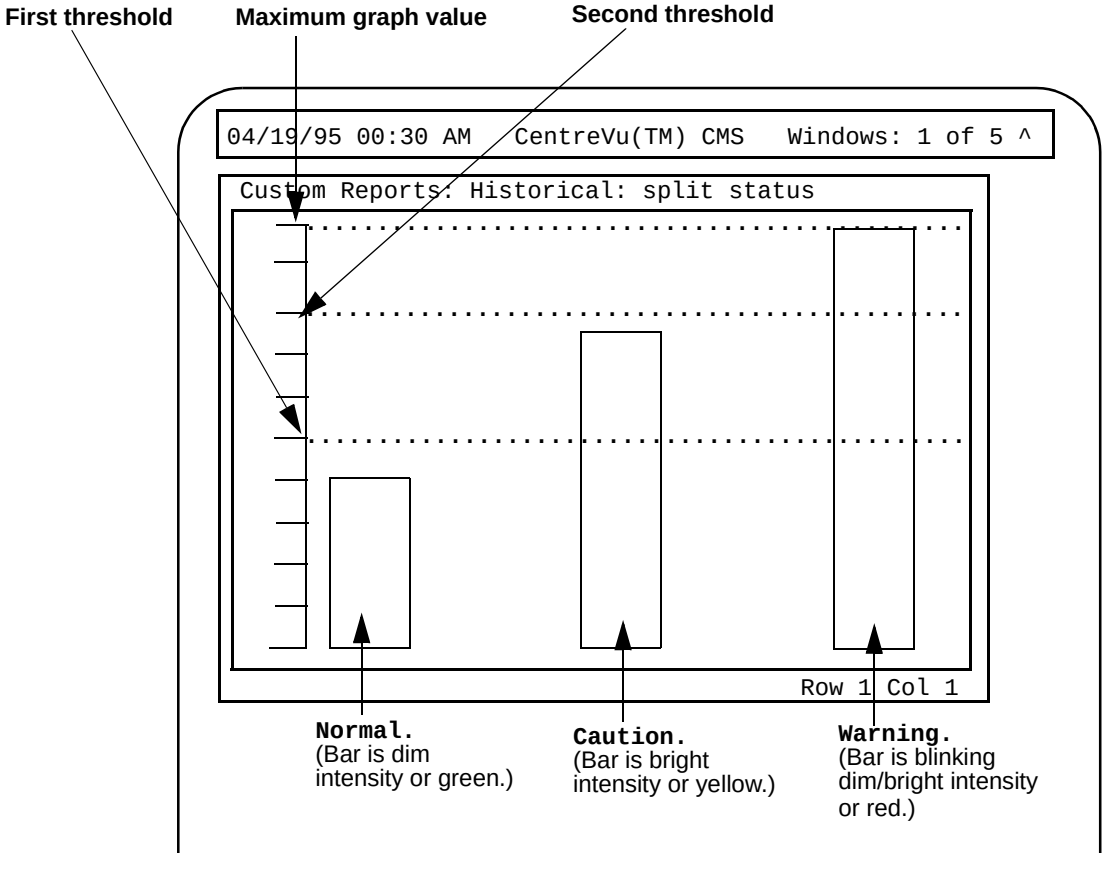


Figure 4-9: Bar Graph Thresholds

Step 7: Define the First Threshold

Enter an **x** in the `First threshold` list to select a fixed threshold, variable threshold, or no threshold. The first threshold identifies the point at or above which the bar changes color — normally to indicate a caution condition (see Figure 4-9). The first threshold should have the lower value of the two thresholds. The value must always be a whole number or decimal.

If you select `None`, the bar does not have a first threshold at which the bar changes color.

If you select `Fixed`, you must also enter, in the accompanying field, the value at or above which the bar changes color.

If you select `Variable name`, you must also enter, in the accompanying field, a variable name. The variable name, which references a report input field, allows the user to enter a first threshold value for the bar when ordering the report. The variable name you enter must be identical to the variable name you assign to a report input field (see "Defining Fields for the Report Input Window" in this chapter).

Step 8: Define the Second Threshold

Enter an **x** in the `Second threshold` list to select a fixed threshold, variable threshold, or no threshold. The second threshold identifies the point at or above which the bar changes color — normally to indicate a warning condition (see Figure 4-9). The second threshold should have the higher value of the two thresholds. The value must always be a whole number or decimal.

If you select `None`, the bar does not have a second threshold at which the bar changes color.

If you select `Fixed`, you must also enter, in the accompanying field, the value at or above which the bar changes color.

If you select `Variable name`, you must also enter a variable name in the accompanying field. The variable name, which references a report input field, allows the user to enter a maximum graph value for the bar when ordering the report. The variable name you enter must be identical to the variable name you assign to a report input field (see the "Defining Fields for the Report Input Window" section in this chapter).

Step 9: Select Normal or Reversed Thresholds

Enter a **y** or **n** to select reversed threshold colors or normal threshold colors. **n** means the bar changes to a caution color at the first threshold and change to a warning color at the second threshold.

Enter **y** if you want to reverse the meanings of the thresholds. With meanings reversed, the bar is a normal color when it is **above** the second threshold. The bar changes to a caution color when the bar is at or below the second threshold but is above the first threshold. The bar changes to a warning color when the bar is at or below the first threshold.

Reversed colors would be appropriate for a bar that represents the percentage of calls answered within service level (see Figure 4-10). In this case, you would want the bar to be a normal color when the percentage is high, a caution color when the percentage goes down, and a warning color when the percentage is very low.

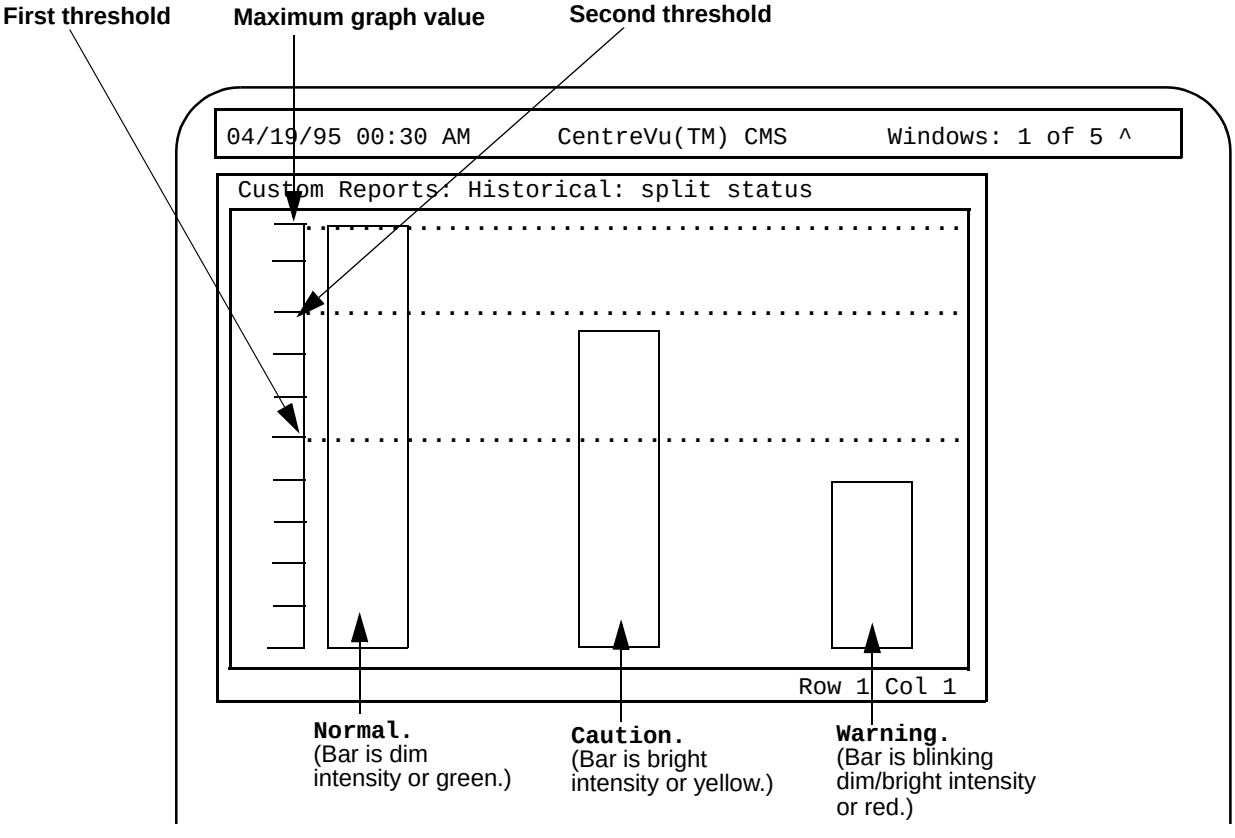


Figure 4-10: Reversed Bar Graph Thresholds

Step 10: Save the Bar Definition

Select the `Save` action list option. → *The Bar window disappears, and* `Successful` *appears in the Screen Painter status line to indicate the bar definition has been added.*

To define additional bars, repeat steps 1 through 10.

Note
You must assign a Row Search ID to the bar before your field definition is complete. When you do, the question mark (?) changes to that ID number. See the "Defining the Rows of Data for a Report" section in this chapter.

Note
If a bar will be a repeated bar (as defined with the Row Search window), you cannot define any other bars in the direction that the bar will repeat. That is, if the bar will be repeated vertically, no other bars (or text or fields) can appear directly below the bar. If the bar will be repeated horizontally, no other bars (or text or fields) can appear directly to the right of the bar.

Changing a Bar Definition

1. On the Screen Painter, place the cursor on the bar you want to change, and select `Bar`. → *The cursor returns to the bar and rests on the lower right corner of the bar. The following message appears on the Screen Painter status line:* `Move cursor to define opposite corner of bar and press RETURN.`
2. If desired, move the cursor using the arrow keys to either make the bar bigger or smaller, and press **Return**. → The Bar window appears.
3. Overtype any data in fields you want to change, and select `Save`. → *The Bar window disappears,* `Successful` *appears in the Screen Painter status line, and the cursor returns to the bar you just changed.*

Defining the Rows of Data for a Report

To complete the definition of a report's fields/bars, you must define the rows of the table(s) that supply data to the fields/bars. To do this, you must:

- Using the Row Search window (Figure 4-11), define the criteria needed to find the appropriate rows of data. Each set of criteria is stored by row search ID.
- Assign the row search ID(s) to the appropriate fields/bars.

Row search criteria are values for either **database items** or **calculations**. In most cases, your criteria will specify variable names rather than specific values for the database items or calculations. These variable names allow *CentreVu* CMS to use the values entered in the Report Input window in its row search criteria. From the rows *CentreVu* CMS finds, report data is retrieved for the report fields. See "How CentreVu CMS Stores and Retrieves Data" in Chapter 1 for a description of this process.

Note

If you copy the design of an existing report, the row search criteria and input fields (as defined in the Define Input window) are copied. If you then delete or change a variable name in the Row Search window, you must delete or change that variable name in the Define Input window.

The task of defining the rows of data for report fields/bars is described in the following pages.

Step 1: Access the Row Search Window

On the Screen Painter, select the `Row search` action list option. → *The Row Search window appears.*

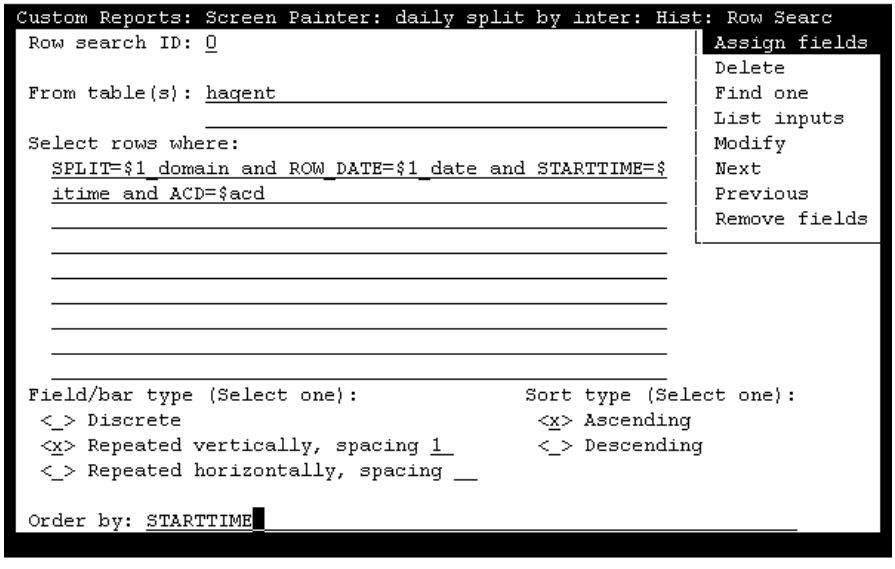


Figure 4-11: The Row Search Window

Step 2: List the Report Input Variables

2a. Select the `List inputs` action list option to see the variable names previously defined on the Define Input window. → *The List Inputs window appears.*

2b. Print the window's contents via the Commands SLK and **`Print window`** option.

2c. Press Exit to return to the Row Search window. → *The List Inputs window disappears and the cursor returns to the Row Search window.*

Note

If you defined report input fields in the Define Input window (or you copied a report design that has report input fields), your row search criteria use the variable names for those input fields. Conversely, to use a variable name in your row search criteria, an input field with that variable name must first be defined in the Define Input window.

Step 3: Select a Row Search ID

Enter a number from 0 to 9 in the `Row search ID` field. The ID identifies the set (or one of the sets) of row search conditions you are using in the report. Thus, for a single report, you can use ten different sets of conditions to select rows of data.

You can create a new set of conditions from scratch or use `Find one` with the `Next/Previous` actions to view and/or change an existing set of conditions. (If you have already assigned row search conditions to a field/bar on the Screen Painter, the row search ID appears as the first character in that field/bar.)

Note

If you are going to use multiple tables, at least one "join" clause must appear in the Row Search ID assigned to the field.

A "join" clause makes the values that *CentreVu* CMS searches on the same in both tables. Therefore, the data extracted from the rows in both tables is related.

A "join" clause has the following format:

```
tablename1.item1 = tablename2.item1
```

where **`item1`** is a database item that the tables have in common. Note that there are different types of "join" clauses.

See Chapter 6, "Advanced Report Design," and your *INFORMIX* documentation for more information on "join" clauses.

Step 4: Select Tables

In the `From table(s)` field, enter the name(s) of the table(s) whose rows supply data. Use a comma to separate multiple table names.

Example:

```
From table(s): hagent,dagent_______
```

If you are going to assign this row search ID to a particular report field/bar, the name(s) in the `From table(s)` field must include the table name(s) that you assigned to the report field.

Note

If a report field merges data from two tables, you must include both table names in this field. See "Selecting Rows from More Than One Table" in Chapter 6 of this manual.

The *CentreVu* CMS database table names are in the following tables.

Table 4-6: Real-Time Table Names

Name	Data Stored
csplit	Split/Skill data for the current interval.
psplit	Split/Skill data for the previous interval.
cagent	Agent data fro the current interval.
pagent	Agent data for the previous interval.
ctkgrp	Trunk group data for the current interval.
ptkgrp	Trunk group data for the previous interval.
ctrunk	Trunk data for the current interval.
ptrunk	Trunk data for the previous interval.
cvector	Vector data for the current interval.
pvector	Vector data for the previous interval.
cvdn	VDN data for the current interval.
pvdn	VDN data for the previous interval.
ccwc	Call Work Code (CWC) data for the current interval.
pcwc	CWC data for the previous interval.

Table 4-7: Historical Table Names

Name	Data Stored
hsplit	Split/Skill data for each intrahour interval.
dsplit	Split/Skill data summarized by day.
wsplit	Split/Skill data summarized by week.
msplit	Split/Skill data summarized by month.
hagent	Agent data for each intrahour interval.
dagent	Agent data summarized by day.
wagent	Agent data summarized by week.
magent	Agent data summarized by month
ag_actv	Agent trace.
haglog	Agent login/logout.
htkgrp	Trunk group data for each intrahour interval.

Table 4-7: Historical Table Names — *Continued*

Name	Data Stored
dtkgrp	Trunk group data summarized by day.
wtkgrp	Trunk group data summarized by week.
mtkgrp	Trunk group data summarized by month.
htrunk	Trunk data for intrahour interval.
dtrunk	Trunk data summarized by day.
wtrunk	Trunk data summarized by week.
mtrunk	Trunk data summarized by month.
hvector	Vector data for each intrahour interval.
dvector	Vector data summarized by day.
wvector	Vector data summarized by week.
mvector	Vector data summarized by month.
hvdn	VDN data for each intrahour interval.
dvdn	VDN data summarized by day.
wvdn	VDN data summarized by week.
mvdn	VDN data summarized by month.
hcwc	CWC data for each intrahour interval.
dcwc	CWC data summarized by day.
wcwc	CWC data summarized by week.
mcwc	CWC data summarized by month.
call_rec	Call record data.
agex	Agent exceptions.
spex	Split exceptions
tgex	Trunk group exceptions.
vecex	Vector exceptions.
vdnex	VDN exceptions.
linkex	Link down exceptions.
mctex	Malicious call trace exceptions.
f_cday	Forecast current day configuration data by split/skill.
f_cdayrep	Current day forecast data by split/skill.

See Appendix A, "Database Items and Calculations", for a description of these tables and the database items they contain.

Step 5: Select Rows in the Table(s)

In the `Select rows where:` field, enter a selection criteria statement to tell *CentreVu* CMS how to select data from the table. The statement specifies value(s) for one or more database items or calculations.

Note

To ensure a reasonable run time for your report, the database items you specify in the `Select rows where:` field should include index items. For a historical report, you should especially include a "where" clause with the ROW_DATE database item. If your row search is based on items that are **not** indexes, your report may take a very long time to run. To check that your row search items are indexes, see the appropriate table listing in Appendix A, "Database Items and Calculations".

Note

Your row search criteria **must always** include the selection of an ACD. See "Where" Clause for Selecting Rows from an ACD" later in this section.

Basic "Where" Clause

A basic clause has the following format:

```
Expression Relational Operator Value
```

The **`Expression`** can be a database item or calculation. The **`Value`** is a whole number. Relational operators available for a "where" clause are as follows:

=	equal to
< > or !=	not equal to
>	greater than
>=	greater than or equal to
<	less than
<=	less than or equal to

Note

Do not try to enter calculation names. They do not work in the `Select rows where` field.

Note

Remember — standard database items consist of upper-case letters only.

As an example of a basic “where” clause, if you are creating a real-time report using data from the Current Interval Split table and you have defined three report fields for the report:

`Split:` (the `SPLIT` database item)

`ACD Calls:` (the `ACDCALLS` database item)

`Average Talk Time:` (the calculation **`ACDTIME/ACDCALLS`**)

Your “where” clause might be:

`Select rows where: SPLIT = 5`

When you run the report, *CentreVu* CMS finds the row in the table (Figure 4-12) for Split 5 and fills in the report fields with data from that row (in bold in the figure).

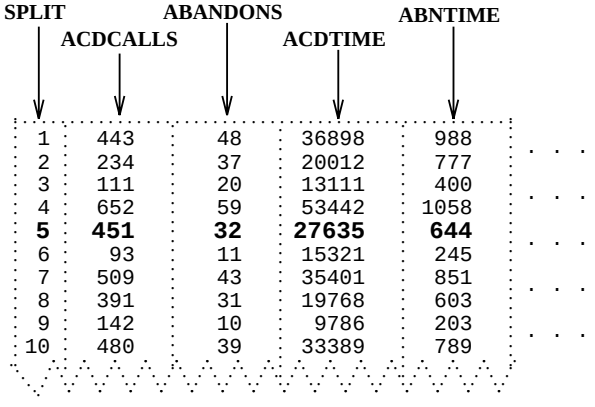


Figure 4-12: Sample Current Interval Split Table

Thus, the report fields show the following data:

`Split: 5`

`ACD Calls: 451`

`Average Talk Time: 61 (the result of` **`27635/451`**`)`

Note

This example of row search criteria would also include the selection of an ACD. See “Where” Clause for Selecting Rows from an ACD” later in this section.

"Where" Clause with Variable

The previous examples create **hardcoded** row search conditions. That is, when you run the report, *CentreVu* CMS always searches for the values you entered in the "where" clause. However, instead of a hardcoded value, you can enter a variable name in a clause. A **variable name** tells *CentreVu* CMS to search for whatever value(s) you or another user choose when you run the report. You must define a variable on the Define Input window before using it in a "where" clause. The variable name then links a report input field to the "where" clause that uses the value(s) entered.

Note

To see the variable names you previously defined on the Define Input window, select the `List inputs` action list option. See "Step 2: List the Report Input Variables."

In a clause with a variable name, you must always enter a dollar sign (**$**) in front of the variable. The format is as follows:

```
Expression Relational Operator $variable
```

As with a basic "where" clause, the **`Expression`** can be a database item or calculation. The relational operators available are the same as those available for a basic "where" clause.

Look at the following "where" clause:

```
Select rows where: SPLIT = $splitvar
```

This "where" clause tells *CentreVu* CMS to search for rows with the Split value the user enters when ordering the report. The example clause presupposes that the variable name, "splitvar," has been defined on the Define Input window.

Multiple "Where" Clauses

To put two or more clauses in a statement, use **`and`** or **`or`**. Use **`and`** to define two or more clauses where *CentreVu* CMS finds only rows that meet all conditions. For example, the following statement searches for rows where splits had an average speed of answer greater than 30 seconds **and** abandons greater than 100.

```
Select rows where: ANSTIME/ACDCALLS > 30 and ABANDONS > 100
```

Use **`or`** to define two conditions where *CentreVu* CMS finds rows that meet either condition but not necessarily both. For example, the following statement searches for rows where splits had too many abandoned calls or too many extension-out calls.

```
Select rows where: ABANDONS>15 or AUXOUTCALLS+ACWOUTCALLS>7
```

"Where" Clause with a Range/List Variable

If a variable name, as defined in the Define Input window, has been assigned the Range/list option, your "where" clause **must** use the equals sign (=) with the variable name.

"Where" Clause for Selecting Rows from an ACD

In your "where" statement, you must always include a clause to select the ACD. If you always want a report to find data for the user's current ACD, append **`and ACD = $acd`** to the "where" statement, as shown in the following example:

```
Select rows where: SPLIT = $splitvar and ROW_DATE =
$datevar and ACD = $acd
```

If you use the variable name **`acd`**, you do not need to define the variable in the Define Input window. *CentreVu* CMS always understands acd to be the current ACD. However, if you wish, you can define a different variable name for the **`ACD`** database item so the user can specify the desired ACD(s) when ordering the report. You can also hardcode the ACD in a "where" clause, as in the following example.

```
Select rows where: ACD = 1
```

"Where" Clause for Excluding Rows of Data

If you want to **exclude** Split 5 from the report, but include all other splits, you might enter a "where" clause as follows:

```
Select rows where: SPLIT != 5
```

or

```
Select rows where: SPLIT <> 5
```

Step 6: Select a Field/Bar Type

Enter an **x** to select one of the following field/bar types.

Discrete

Select the Discrete type if *CentreVu* CMS will find only one value for each report field/bar to which you are assigning this Row Search ID. *CentreVu* CMS will find only one value if the field/bar is an aggregate function (SUM, AVG, MIN, or MAX), or if **both** of the following conditions are true:

- You enter a "where" statement that is so specific that *CentreVu* CMS finds only one row.
- On the Define Input window, you select **n** for Range/list for all variables used in the "where" statement. For example, if you select the Daily Split (**dsplit**) table and enter the following "where" statement:

  Select rows where: ROW_DATE = $datevar and SPLIT = $splitvar and ACD = $acd

  and you select **n** for Range/list for both the splitvar and **datevar** variables, *CentreVu* CMS will find a single row containing the date and split the user enters when ordering the report.

Note

If you are assigning the row search ID only to fields and bars that contain aggregate functions (sum, max, min, or avg), you will normally select the Discrete type. See "Repeating Aggregate Function Values" in Chapter 6 of this manual.

Repeated vertically

Select Repeated vertically to display a column of multiple field values or a vertical series of bars, one for each value.

If you select Repeated vertically, you must also enter a number in the **Spacing** field. This number tells *CentreVu* CMS how many lines to go down to display each value. **1** means to display a value on every line. **2** means to display a value on every other line.

Repeated horizontally

Select Repeated horizontally to display multiple field values in row format or a horizontal series of bars, one for each value.

If you select `Repeated horizontally`, you must also enter a number in the `Spacing` field. This number tells *CentreVu* CMS how many characters to move horizontally from the **beginning** of one value to the beginning of the next. This means that the spacing you enter must include the blank spaces between fields/bars **and** the width of a field/bar. For example, if a field/bar is four characters wide and you enter **`8`** in the Spacing field, *CentreVu* CMS displays each value with four blank characters in between.

Repeated values, whether repeated horizontally or vertically, apply only if *CentreVu* CMS finds multiple values for the report field(s). *CentreVu* CMS finds multiple values if **one** of the following conditions exists:

- Your "where" statement specifies one or more ranges of values.
- On the Define Input window, you entered **`y`** for Range/list for at least one variable.
- Your "where" statement is general enough to select multiple rows.

  One example of a general "where" statement for the **`dsplit`** table would be:

  ```
  Select rows where: ROW_DATE = $datevar and ACD
  = $acd
  ```

  Since no split values are specified, *CentreVu* CMS will display, for a user-specified date, a value for every split in the ACD.

If, based on your "where" statement and Define Input entries, *CentreVu* CMS finds multiple values, but you select `Discrete`, the report displays only the first value *CentreVu* CMS finds for each report field.

Note

If you select `Repeated vertically`, you cannot define any fields directly under a repeated field. If you select `Repeated horizontally`, you cannot define any fields directly to the right of a repeated field.

Step 7: Select a Sort Type

Enter an **x** to select either ascending or descending order for values of the database item(s) in the `Order by` field. Ascending order means *CentreVu* CMS displays data from the lowest to the highest values. Descending order means *CentreVu* CMS displays data from the highest to the lowest values.

Continuing with the previous example, if the `Order by` database item is **ROW_DATE** and you choose a descending order, the data will be listed as shown below.

```
DATE       SPLIT   ACDCALLS
10/12/92   1       7
10/12/92   2       6
10/12/92   3       5
10/11/92   1       25
10/11/92   2       50
10/11/92   3       41
10/10/92   1       40
10/10/92   2       36
10/10/92   3       30
```

Figure 4-13: Sample of Data Ordered by Date in Descending Order

For dates and times, lowest values are those dates/times farthest in the past.

Step 8: Select a Sort Order for Data

Enter a database item (with its table name) in the `Order by` field to specify how multi-row data should be sequenced in the report. *CentreVu* CMS orders the rows of data by the values for the database item. For example, you might enter **`dsplit.ROW_DATE`** in the `Order by` field. *CentreVu* CMS would then display data by date, as in the following report.

```
DATE       SPLIT   ACDCALLS
10/10/92   1       40
10/10/92   2       36
10/10/92   3       30
10/11/92   1       25
10/11/92   2       50
10/11/92   3       41
10/12/92   1       7
10/12/92   2       6
10/12/92   3       5
```

Figure 4-14: Sample of Data Ordered by Date in Ascending Order

If, on the other hand, you enter **`dsplit.SPLIT`** in the `Order by` field, *CentreVu* CMS displays data by split, as shown in the following report.

```
DATE        SPLIT   ACDCALLS
10/10/92      1       40
10/11/92      1       25
10/12/92      1        7
10/10/92      2       36
10/11/92      2       50
10/12/92      2        6
10/10/92      3       30
10/11/92      3       41
10/12/92      3        5
```

Figure 4-15: Sample of Data Ordered by Split

If you leave the `Order by` field blank, *CentreVu* CMS displays data in the report using the same sequence *CentreVu* CMS uses to store the data in the table.

For historical reports only, you can enter more than one database item in the **`Order by`** field. The database items must be separated by commas. The effect of more than one sorting database item is that data will be ordered first by values found for the first sorting database item, then sorted by values for the second sorting database item, and so on.

For example, `Order by:` **`dsplit.SPLIT,dsplit.ACDCALLS`** would cause the data shown in the previous illustration to be ordered as follows:

DATE	SPLIT	ACDCALLS
10/12/92	1	7
10/11/92	1	25
10/10/92	1	40
10/12/92	2	6
10/11/92	2	36
10/10/92	2	50
10/12/92	3	5
10/11/92	3	30
10/10/92	3	41

Figure 4-16: Sample of Data Ordered by Split and ACD Calls

Notice that the data is first sorted by split number, then by the number of ACD calls. As a result, the dates appear totally out of sequence.

Step 9: Save Your Row Search Conditions

Select the `Modify` action list option. → `Successful` *appears in the window's status line to indicate the Row Search conditions have been added.*

After you have successfully saved your row search conditions, you can then assign the row search ID to report fields/bars.

Assign a Row Search ID to Report Field(s) and Bar(s)

To complete the definition of data for a field or bar, you must assign a row search ID to the field/bar. Use the following steps to do this.

Note

If a field already has an assigned row search ID (for example, a field that is part of a design you have copied), you must first **remove** the existing row search ID assignment. See "Changing the Row Search-Field Assignment(s)" later in this section.

1. On the Screen Painter, place the cursor on a field to which you want to assign a row search ID. To assign a row search ID to several fields at one time, place the cursor in a position from which you can define an appropriate block.
2. Select the `Row search` action list option. → *The Row Search window appears.*
3. Enter the desired row search ID and select the **`Find one`** action list option. → *The row search conditions for the ID appear.*
4. Select the `Assign fields` action list option. → *The Row Search window disappears, the cursor moves to its original position* `Move cursor to define a block and press RETURN` *appears on the status line.*

5. Move the cursor using the Tab, Shift Tab, or arrow keys. If the block you define contains one or more characters of a field/bar, *CentreVu* CMS assigns the row search ID to that field/bar.
   → *The cursor defines a block on your terminal in inverse video (or color).*
6. Press Return.
   → *The ? in each field/bar changes to the row search ID to indicate the row search conditions have been assigned. Additional characters fill the field or bar as shown in the following illustrations.*

If your row search conditions specify discrete fields/bars, assigned fields appear with lowercase x's and assigned bars appear with uppercase X's (see Figure 4-17).

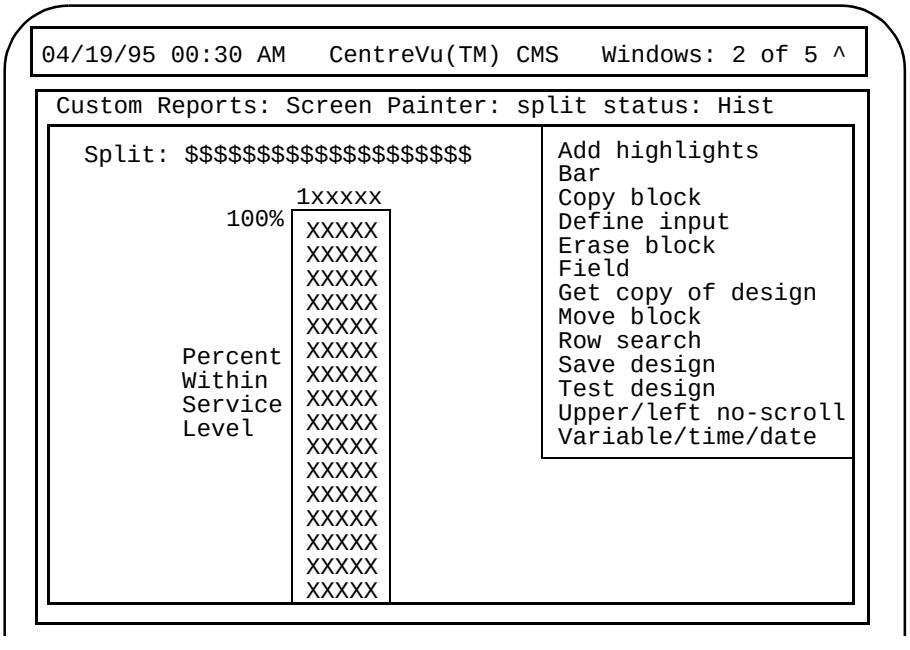


Figure 4-17: Sample Discrete Field and Bar Design

A report with the design in Figure 4-17 might appear as shown in Figure 4-18.

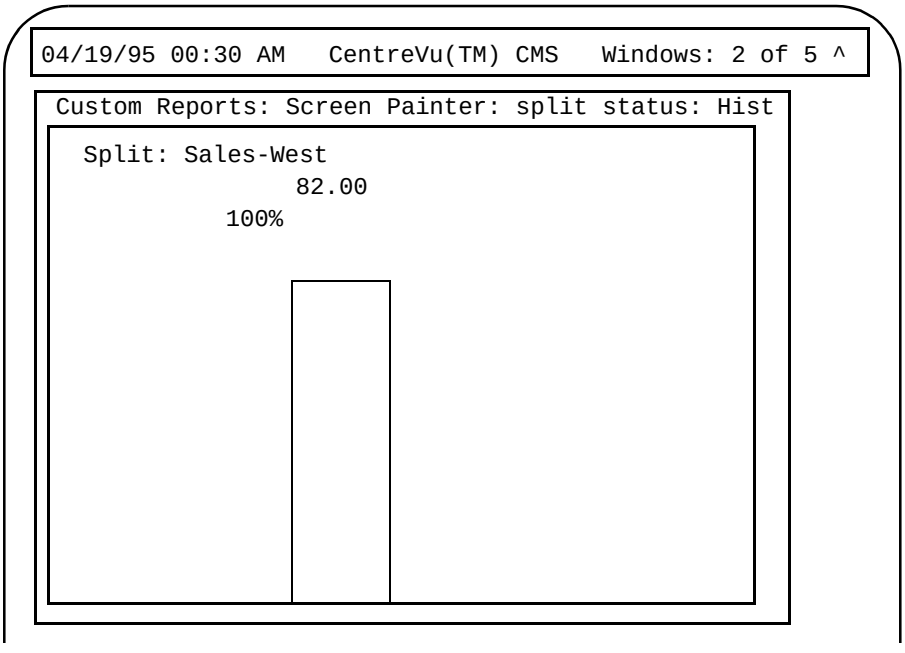


Figure 4-18: Sample Report with Discrete Field and Bar

If your row search conditions specify vertically repeated fields/bars, assigned fields appear with lowercase v's and assigned bars appear with uppercase V's (see Figure 4-19).

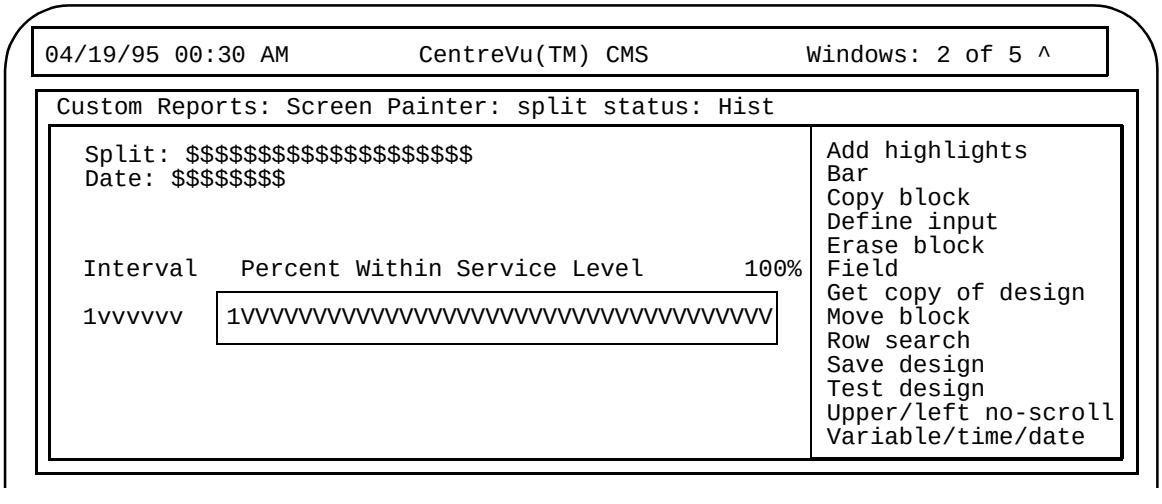


Figure 4-19: Sample Field and Bar Design — Vertically Repeated

A report with the design in Figure 4-19 might appear as shown in Figure 4-20.

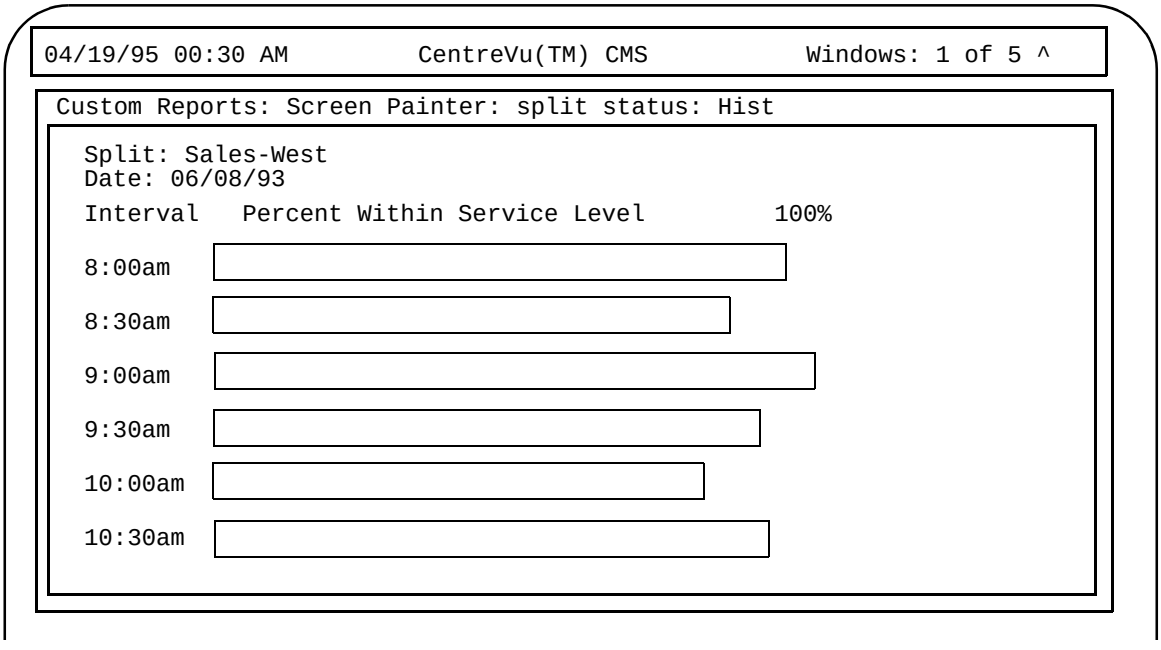


Figure 4-20: Sample Report with Vertically Repeated Field and Bar

If your row search conditions specify horizontally repeated fields/bars, assigned fields appear with lowercase h's and assigned bars appear with uppercase H's (see Figure 4-21).

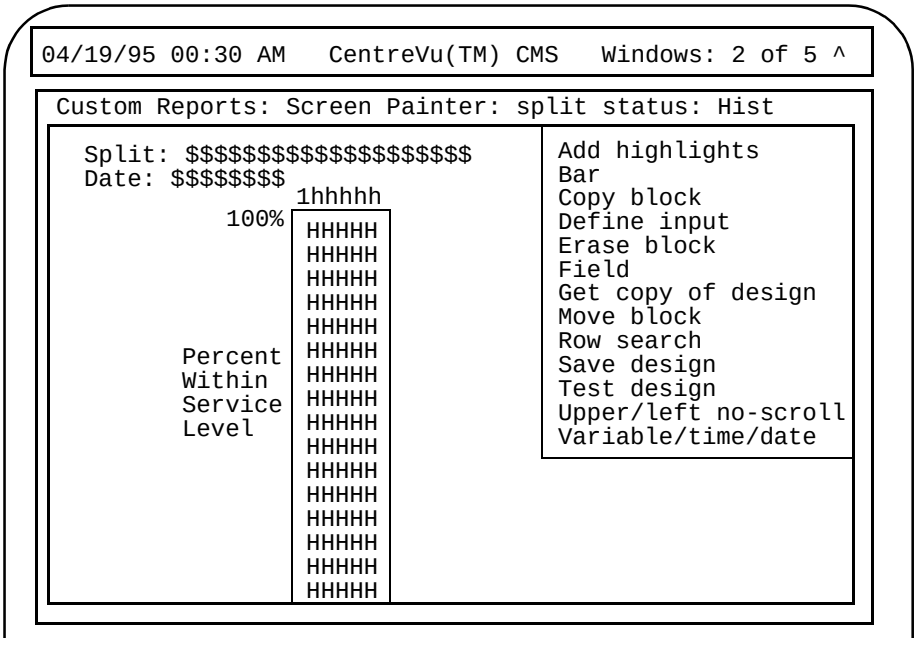


Figure 4-21: Sample Field and Bar Design — Horizontally Repeated

A report with the design in Figure 4-21 might appear as shown in Figure 4-22.

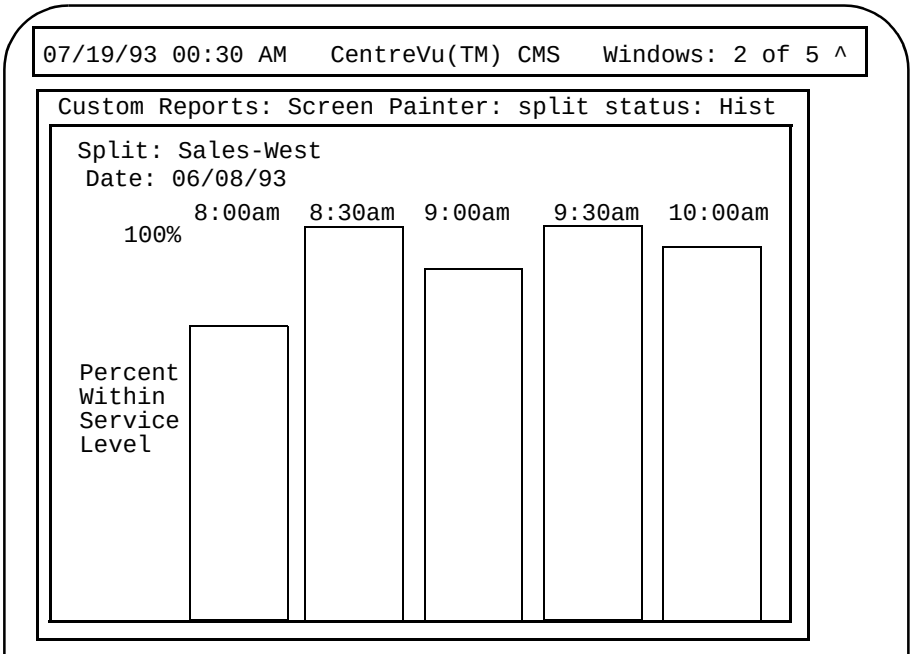


Figure 4-22: Sample Report with Horizontally Repeated Field and Bar

Changing Row Search Conditions

1. On the Screen Painter, select the `Row search` action list option. → *The Row Search window appears.*
2. Enter a number from 0 to 9 in the `Row Search ID` field, and select the `Find one` action list option. → *The row search conditions for the ID appear.*

   Note
   For a `Find one` search, only your entry in the `Row search ID` field counts. *CentreVu* CMS ignores the other fields.

3. Change data in any field(s), and select the `Modify` action list option. → `Successful` appears in the window's status line to indicate the row search conditions have been changed.

The changes apply to any fields to which you had previously assigned the Row Search ID.

Changing the Row Search-Field Assignment(s)

To change the row search ID assigned to a particular field, you must first disassociate the field from the row search ID. You do this using the `Remove fields` action list option.

1. On the Screen Painter, place the cursor on a field/bar from which you want to disassociate the row search ID. To disassociate a row search ID from several fields/bars at one time, place the cursor in a position from which you can define a block that includes all appropriate fields/bars.
2. Select the `Row search` action list option. → *The Row Search window appears.*
3. Enter the desired row search ID and select the `Find one` action list option. → *The row search conditions for the ID appear.*
4. Select the `Remove fields` action list option. → *The Row Search window disappears, the cursor moves to its original position on the Screen Painter, and* `Move cursor to define a block and press RETURN` *appears on the status line.*
5. Move the cursor using the Tab, Shift Tab, or arrow keys. The block you define should contain all characters of each field/bar from which you are disconnecting the row search ID. → *The cursor defines a block on your terminal in inverse video (or color).*
6. Press Return. → *The row search ID in each field/bar is replaced by a question mark ? to indicate the row search conditions have been disconnected.*
7. Assign a new row search ID to the fields/bars.

Defining Fields to Show Run Time/Date and User Inputs

The time or date when a report is run is not stored in any database table. Instead, *CentreVu* CMS simply recognizes when it has finished processing the report, and if requested, displays this information on the report. Similarly,

CentreVu CMS knows what your current ACD is when you order the report and can display the current ACD name or number on the report.

To display the report's run-time, run-date, or the current ACD when you ordered the report, you must define a field using the Var/Time/Date window (Figure 4-23). The Var/Time/Date window also allows you to define fields that mirror your entries in the Report Input window. That is, *CentreVu* CMS simply takes the values you enter for an input variable (as defined in the Define Input window) and displays those values on the report.

To define a field with one of these types of information, use the following steps.

Step 1: Access the Variable/ Time/Date Window

On the Screen Painter, place the cursor where the left end of the field should appear, and select `Variable/time/date`.

→ *The Var/Time/Date window appears.*

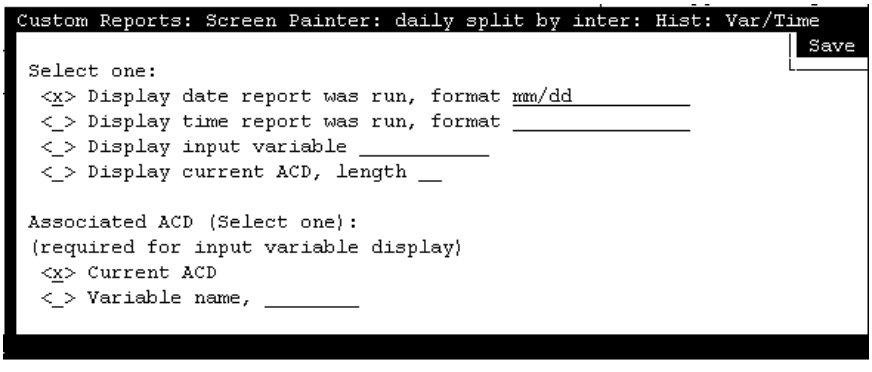


Figure 4-23: The Var/Time/Date Window

Step 2: Define the Display Type and Format

Note

You are given flexibility in defining the appearance of the date and time fields; however, only text matching the defined formats is converted into their equivalents. Any other text is simply displayed as entered.

2a. Enter an **x** to select a display type. You must also complete the associated field.

The display options are:

date report was run

Select this option to display the date you ran the report. You must also specify a date format with appropriate punctuation in the associated field. You can enter any of the following formats or combination of formats:

Table 4-8: Date Formats

mm	The numerical month.
MMM	The month represented by three letters.
yy	The year as two digits.
yyyy	The year as four digits.
dd	The numerical day of the month.
jjj	The day of the year in the Julian calendar.
www	The day of the week as three letters.

An example of combined formats would be the standard date format **mm/dd/yy**. Or, you could enter just the day and month **mm/dd**.

time report was run

Select this option to display the time you ran the report. You must also enter a time format with the appropriate punctuation.

You can use any of the following formats:

Table 4-9: Time Formats

HH	The hour only, in military time (24-hour clock).
hh	The hour only, according to a 12-hour clock.
mm	The number of minutes after the hour only.
ss	The number of seconds into the minute.

Table 4-9: Time Formats

HH:mm:ss or HH:mm	Military time, either to the second or to the minute.
hh:mm:ssam or hh:mmam	12-hour clock time, with AM or PM attached, either up to the second or up to the minute.

2b. Verify that the format of your date and time entries are acceptable by accessing the help screen for both the date and time fields.

input variable Select `input variable` to display data exactly as it is entered in one of the fields on the Report Input window. You must also enter, in the associated field, the variable name assigned to the report input field (on the Define Input window).

current ACD Select `current ACD` to display the number or name of the ACD that was current when you ordered the report. You must also enter, in the associated field, the number of characters, 1 to 20, that the ACD field should have. If you have defined names in the Dictionary subsystem for your ACD(s), then you may want to enter **20**, since this is the maximum length for Dictionary names.

Step 3: Save the Var/Date/Time Field

Select `Save`. → *The Var/Time/Date window disappears. **CentreVu** CMS marks the field in one of the following ways:*

For a **`date`**, *the format you defined is displayed (for example,* **`mm/dd/yy`**).

For a **`time`**, *the format you defined is displayed (for example,* **`hh:mmam`**).

For an **`input variable`**, *a string of dollar signs* (**`$$$`**) *is displayed, with the number of dollar signs equaling the field length you assigned to the variable's input field on the Define Input window.*

For **`current ACD`**, *a string of dollar signs* (**`$$$`**) *is displayed, with the number of dollar signs equaling the length you specified in the Var/Date/Time window.*

Repeat Steps 1 through 3 for each field you want to define.

Saving Your Work

To save everything you have done so far, select `Save design` from the Screen Painter. If you have defined something incorrectly, an error message will inform you of this. You must go back and correct any errors before saving.

5 Completing Your Custom Report

Highlighting Fields

For each individual *CentreVu*™ Call Management System (*CentreVu* CMS) custom report, you can administer how the screen displays the colors and brightness levels of the text, data and background. You can also administer color, brightness level, reverse video, and underlining to emphasize individual fields and text in the report.

Note

You **cannot** change the way bars are displayed with the procedure that follows. However, you **can** change the bars' threshold colors via the **Commands** SLK and the `Options: Color` submenu selection.

To change the way the screen displays fields and text in your report, do the following steps:

1. On the Screen Painter, place the cursor in a position where you want one corner of a block to be, and select `Add highlights`.
   → *The cursor returns to its original position, and the following message appears on the status line:* `Move cursor to define opposite corner of block, press RETURN.`
2. Move the cursor to a position where you want the opposite corner of the block to be. The block should include all field text you want to highlight.
   → *The block you are defining becomes highlighted.*
3. Press **Return**.
   → *The Add Highlights window appears.*

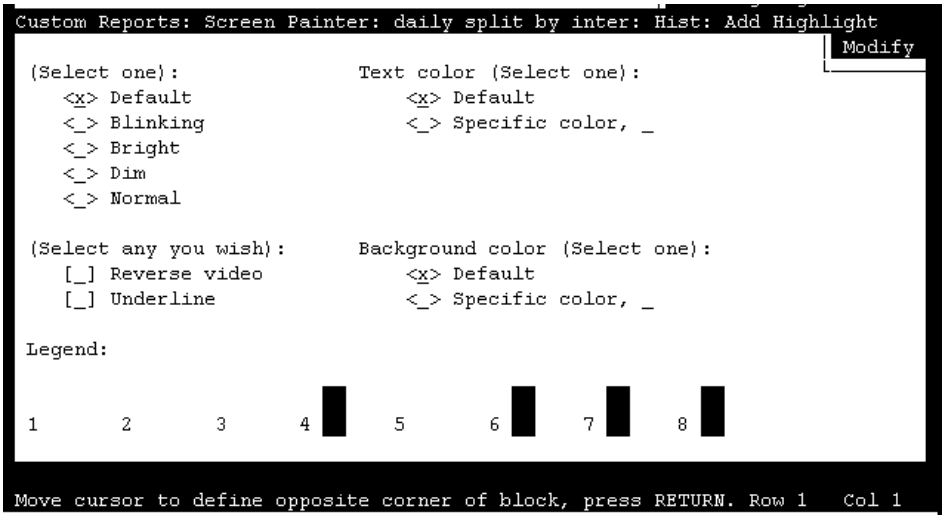


Figure 5-1: The Add Highlights Window

4. Select the highlighting options you want. The highlighting options are:

   Select one:

   - **Default** Reflects the text and background settings administered for this terminal.
   - **Blinking** Causes the text/data in the defined area to blink (continuous alternation between dim and normal brightness or foreground/ background colors).
   - **Bright** Makes the text/data in the defined area become bright. If your terminal does not have a “bright” capability, “bright” will equal “normal.”
   - **Dim** Makes the text/data in the defined area become dim. If your terminal does not have a “dim” capability, “dim” will equal “normal.”
   - **Normal** Makes the text/data in the defined area a brightness level between “dim” and “bright.”

Select any you wish:

- **Reverse video** — This setting causes brightness/color settings for text and background to be reversed. For example, for an amber monochrome terminal, reverse video makes the background amber and the text/data black. For a color terminal set for white text on blue background, reverse video makes the background white and the text blue.

  The brightness and blinking options also affect the display of reverse video.

- **Underline** — This setting draws a line under any text/data in the defined area. If you also select “Reverse video," the underline color/brightness is reversed.

Text color (for color terminals only):

- **Default** — This setting displays text in the color set in the Options: Color window, which you access via **Commands**.
- **Specific color** — This setting assigns to the text one of the eight colors displayed at the bottom of the window. If you select this option, you must enter the number of the color in the associated field you want.

Background color (for color terminals only):

- **Default** — This setting displays the background in the color set in the Options: Color window, which you access via **Commands**.
- **Specific color** — This setting assigns to the background one of the eight colors displayed at the bottom of the window. If you select this option, you must enter, in the associated field, the number of the color you want.

5. Select the `Modify` action list option. → *The Add Highlights window disappears.*

Note
For some color terminals, dim or bright settings may affect the colors selected.

Defining Stationary (No-Scroll) Areas

Many standard reports are bigger than the available space on your terminal. As a result, when you display the report on your terminal, you must scroll the window to see all the data, either down or to the side. However, you will notice that some report text or data fields do not move. These stationary, or no-scroll, parts of the report normally are column headers, column totals, and row identifiers.

In the following report illustration, the shaded areas are no-scroll areas.

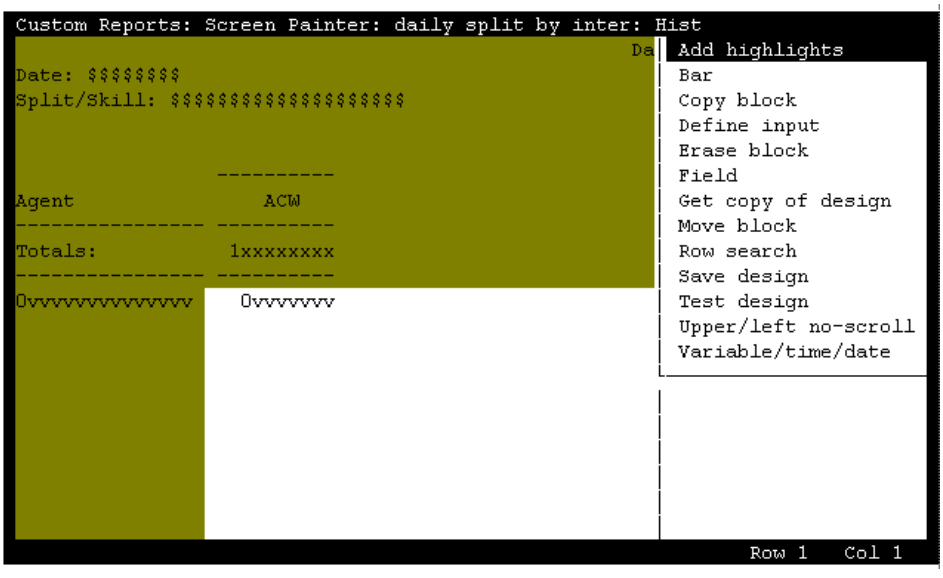


Figure 5-2: Sample Non-Scroll Areas

No-Scroll areas apply only to reports that you are viewing on your terminal. If you display the report on your terminal, almost half of the right side of the report is hidden from view. Thus, you have to scroll to the right and data on the left disappears off the left side of the window. But while you scroll the data, the agent on the left remain in place so you always know what agent the data is for.

And if you have over 20 agents listed in the report in Figure 5-2, some rows of data at the bottom will be hidden. Thus, you have to scroll down to see the rows of data past 20 agents. Meanwhile, data at the top scrolls off the top of the window. But the report title, the column headers, and the `Totals:` row stay in place.

In your custom report, you may wish to assign areas as no-scroll areas. You can assign one no-scroll area on the left side of the report and one no-scroll area at the top of the report. To define no-scroll areas, use the following steps:

1. On the Screen Painter, select `Upper/left no-scroll`. → *The cursor returns to the upper left corner of the report. The following message appears on the status line:* `Move the cursor down and right. Press RETURN when finished.`
2. To define a no-scroll area at the top, move the cursor down the desired number of lines. To define a no-scroll area on the left, move the cursor to the right. → *CentreVu CMS highlights the no-scroll area as you move the cursor.*
3. Press **Return** to save the no-scroll area.

Changing a No-Scroll Area

1. On the Screen Painter, select `Upper/left no-scroll`. → *The cursor returns to a position at the edge of the top and/or left no-scroll areas. The following message appears on the status line:* `Move the cursor down and right. Press RETURN when finished.`

2. To change a no-scroll area at the top, move the cursor up or down the desired number of lines. To change a no-scroll area on the left, move the cursor to the right or left. → *As you move the cursor, CentreVu CMS highlights the no-scroll area that you add or removes the highlight for the no-scroll area you delete.*

3 Press **Return** to save your changes.

Saving the Report Design

Before you exit the Screen Painter, you must save your report design. If you do not save the design, all the work you did since accessing the Screen Painter will be lost, **including** any work you did with secondary windows. You must save the design, even if you successfully test it with the `Test design` action list option.

You can save a design that does not test successfully because there are errors. **Do not** run a report design until you have corrected any errors and it tests successfully.

To save your report design, do the following:

1. On the Screen Painter, select `Save design`. → `Successful` *may appear on the Screen Painter status line.*

   If you did not define all necessary elements of the design, the Save Design window appears with a list of errors.

   When there are errors in the design, certain items may be deleted from the design and the next time the design is read into the screen painter, these elements will not appear.

2. After you save your design, you can continue working on the design, test it, or exit the Screen Painter and continue working on the design at a later time.

   If *CentreVu* CMS displays errors, do the following:

   - Note the errors
   - Press the Exit SLK
   - Fix the errors on the Screen Painter
   - Repeat Step 1.

   Note

   If you exit the Screen Painter without first fixing errors found in the Save Design window, then the report items associated with the errors will be deleted from the Screen Painter.

Testing the Report Design

Instead of running a report and having it fail because you had errors in your design, you can test the design directly from the Screen Painter. Testing your report design regularly can save you time.

Use the following steps to test a report design:

1. On the Screen Painter, select `Test design`. → *If your design has Phase One errors, a window (Figure 5-3) listing those errors appears. See the following section, "Test Design Error Messages" for a description of these errors. If your design has no Phase One errors, the Test Design window (Figure 5-4) appears with a facsimile of the Report Input window.*

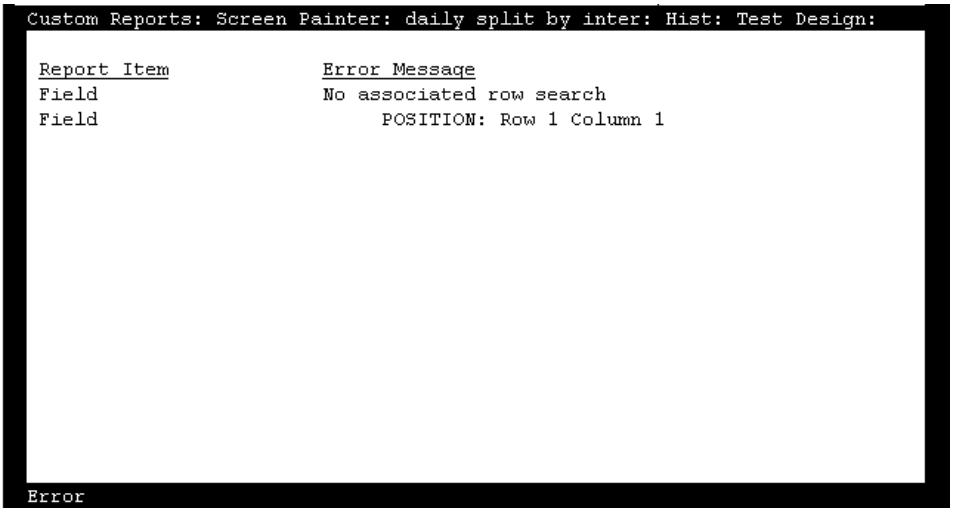


Figure 5-3: Sample Test Design Window - Phase One Errors

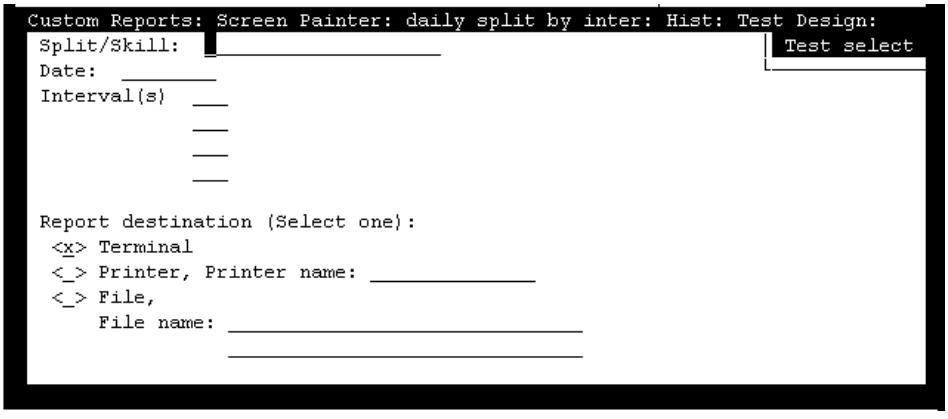


Figure 5-4: Sample Test Design Input Window

2. Complete the fields on the Report Input window, and select `Test select`.

→ *A list of the row search conditions appears (Figure 5-5), with a list of any Phase Two errors. See "Test Design Error Messages", for a description of these errors.*

Note

If you had Phase One errors, fix the errors indicated, and go back to Step 1.

If you have no errors `No errors found` *appears in the window.*

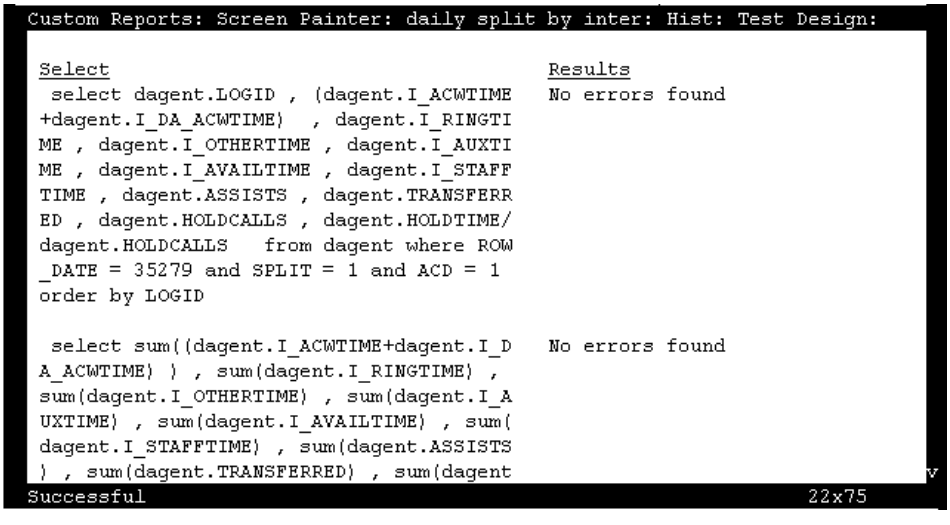


Figure 5-5: Sample Test Design Input Window

3. If you have errors, then do the following:
   - Note the errors
   - Press the Exit SLK
   - Fix any errors
   - Select `Test design` again
   - Repeat Steps 1 and 2.

   Note
   For real-time reports the link to the switch must be up for the current ACD, and there must be data for the particular item you are selecting, such as Split 5.

If you have no errors, press the Exit SLK and continue with your design, or save the design and exit the Screen Painter.

Test Design Error Messages

Test design error messages are divided into three categories:

- Phase 1 Error Messages (detected before the sample input screen appears)
- Phase 2 Historical Error Messages (require the sample input before occurring)
- Phase 2 Real-Time Error Messages (require the sample input before occurring).

Phase 1 Error Messages

This section contains Phase 1 error messages. Table 5-1 lists the messages alphabetically, and includes a cause and a recommended solution for each message.

Table 5-1: Phase 1 Error Messages

Message:	`$<variable name> not defined`
Cause:	The “where” clause contains a variable that is not defined.
Solution:	Define the variable using the define input action or remove the variable from the row search criteria.
Message:	`Cannot mix aggregates and nonaggregates in the select`

Table 5-1: Phase 1 Error Messages — *Continued*

Cause:	You cannot specify both aggregate columns and nonaggregate columns in the same select for real-time reports.
Solution:	Create two identical row search conditions and apply one to the aggregate columns and one to the nonaggregate columns.
Message:	`Cannot use the SYN function for order by`
Cause:	You cannot use a synonym to sort the output.
Solution:	Remove the SYN aggregate from the `Order by` field.

Table 5-1: Phase 1 Error Messages (Contd)

Message:	`CMS system error - Check the error log`
Cause:	A *CentreVu* CMS system error occurred while the select executed. The error should be recorded in the error log.
Solution:	Check the error in the error log to initiate corrective action.

Caution

If you run a report that merges data from two tables (particularly tables with large amounts of data) into a single field and your **`Select rows where`** statement is not specific enough, you may get this error message. The specific cause may be that the number of selected rows is very large, and *CentreVu* CMS does not have enough space to create temporary files. If this is the case, you should add additional “where” clauses to the row search criteria.

Message:	`CMS system error - Data collection off`
Cause:	*CentreVu* CMS cannot test the row search criteria while data collection is off.
Solution:	Turn data collection on and rerun test of report design.
Message:	`CMS system error - Too much data retrieved - try a more restrictive search`
Cause:	Too much data was retrieved with the given row search criteria.
Solution:	Add more conditions to the row search criteria so that fewer rows are retrieved.

Message:	`CMS system error - Updating translations`
Cause:	*CentreVu* CMS cannot test the row search criteria while *CentreVu* CMS is receiving the set of configuration data from the switch.
Solution:	Wait until configuration data has been sent. Then rerun the test of report design.

Table 5-1: Phase 1 Error Messages (Contd)

Message:	`keyword AVG invalid in where clause`
Cause:	You cannot use the keyword AVG in row search criteria.
Solution:	Remove the keyword AVG from the row search criteria.
Message:	`keyword BETWEEN invalid for real-time`
Cause:	You cannot use the keyword BETWEEN in row search criteria for a real-time report.
Solution:	Remove the keyword BETWEEN from the row search criteria.
Message:	`keyword COUNT invalid in where clause`
Cause:	You cannot use the keyword COUNT in row search criteria.
Solution:	Remove the keyword COUNT from the row search criteria.
Message:	`keyword MAX invalid in where clause.`
Cause:	You cannot use the keyword MAX in row search criteria.
Solution:	Remove the keyword MAX from the row search criteria.
Message:	`keyword MIN invalid in where clause.`
Cause:	You cannot use the keyword MIN in row search criteria.
Solution:	Remove the keyword MIN from the row search criteria.
Message:	`keyword SUM invalid in where clause.`
Cause:	You cannot use the keyword SUM in row search criteria.
Solution:	Remove the keyword SUM from the row search criteria.
Message:	`keyword SYN invalid in where clause.`
Cause:	You cannot use the keyword SYN in row search criteria.
Solution:	Remove the keyword SYN from the row search criteria.

Phase 2 Historical Error Codes

This section contains the *INFORMIX** error codes most likely to appear for *CentreVu* CMS historical reports. Each code includes a description of the error and the recommended corrective action.

These errors are reported in the following format:

```
INFORMIX error: <error number>
```

In addition, a circumflex (^) may appear in the listed Select statement(s) to mark the location of an error.

Note

If an error code appears that is not listed in this document, see the *INFORMIX SQL Relational Database Management System Reference Guide* for *INFORMIX* SQL Version 4.10.

Table 5-2: Phase 2 Historical Error Codes

—201	**Description:** A syntax error has occurred. **Solution:** Check that you have not misspelled an RDSQL statement, placed key words out of sequence, or included an *INFORMIX*-SQL reserved word in your query.
—202	**Description:** An illegal character has been found in the statement. **Solution:** Remove the illegal character (often a nonprintable control character) and resubmit the statement
—203	**Description:** An illegal integer has been found in the statement. **Solution:** Integers must be whole numbers from -2,147483,647 to 2,147,483,647. Check that you have not included a number with a fractional portion or a number outside the acceptable range. Check also that you have not inadvertently entered a letter in place of a number (for example, `125p3` instead of `12503`).
—204	**Description:** An illegal floating-point number has been found in the statement. **Solution:** Check that you have not inadvertently entered a letter in place of a number (for example, `125p3` instead of `125.03`).

*INFORMIX is a registered trademark of Informix Software, Inc.

Table 5-2: Phase 2 Historical Error Codes — *Continued*

—206	**Description:** The specified table name is not in the database. **Solution:** Check the spelling of the table name in your statement.
—217	**Description:** Column *column-name* not found in any table in the query. **Solution:** Correct the spelling of the database item and ensure that the item exists in the database table. Check for the presence of required commas and quotes.
—219	**Description:** Wildcard matching may not be used with noncharacter types. **Solution:** Wildcards (*, ?) and characters enclosed in brackets [ ] can be used only with CHAR data types. Check the data type for the offending column.
—220	**Description:** There is no FROM clause in the query. **Solution:** Must include a FROM clause in the query. Check that you do not have an illegal character ($, #, &, etc., or a CONTROL character) in the line prior to the FROM keyword.
—223	**Description:** Duplicate table name *table-name* in the FROM clause. **Solution:** Remove the redundant table name from the statement or use an alias to rename one of the tables.
—228	**Description:** cannot have negative characters. **Solution:** Check that you have not included a negative CHAR data type (for example, -a or -p) in your statement.
—278	**Description:** Too many ORDER BY columns; maximum is eight. **Solution:** Reduce the number of columns included in the ORDER BY clause to eight or less.

Table 5-2: Phase 2 Historical Error Codes — *Continued*

—280	**Description:** Total size of ORDER BY columns exceeds 120 bytes. **Solution:** Reduce the number of columns included in the ORDER BY clause so that the total number of characters is less than or equal to 120 (perhaps delete a CHAR column of 30 or more characters).
—282	**Description:** Found a quote for which there is no matching quote. **Solution:** Check that all quoted strings are properly terminated with a quote.
—284	**Description:** A subquery has not returned exactly one value. **Solution:** Check data for the subquery. Restructure the subquery by adding more components in the WHERE clause so that only one value is returned.
—297	**Description:** The SELECT list may not contain a subquery. **Solution:** Remove the subquery from the SELECT list in the statement.
—300	**Description:** There are too many GROUP BY columns (maximum is eight). **Solution:** Reduce to eight or less the number of nonaggregate database items that are assigned the same row search ID as that assigned to an aggregate function.
—301	**Description:** The total size of the GROUP BY columns exceeds 120 characters. **Solution:** The total number of characters in all columns listed in the GROUP BY list exceeds 120 characters. Reduce the number of nonaggregate database items that are assigned to a row search ID that is also assigned to an aggregate function.
—303	**Description:** Expression mixes columns with aggregates. **Solution:** Restructure your query so that columns and aggregates are not included in the same expression.

Table 5-2: Phase 2 Historical Error Codes — *Continued*

—309	**Description:** ORDER BY database item must be included in a report field to which the row search ID is assigned. **Solution:** Check that database items included in the ORDER BY clause appear in the report and are assigned to row search ID.
—324	**Description:** Ambiguous database item. **Solution:** A database item in your row search criteria exists in more than one table also cited in your row search criteria. Prepend each database item with the appropriate table name.
—352	**Description:** Database item not found. **Solution:** Check the spelling of the database item.
—367	**Description:** Sums and averages cannot be computed for character columns. **Solution:** Check that you have not included a database item of a string type (VDN, LOGID, etc.) in the aggregate function statement.
—522	**Description:** A database item in a field/bar does not exist in the table specified in the field's row search ID. **Solution:** Check the Select statement that has the error. The database item that does not exist in the table will be marked with a circumflex (^). Change or delete the database item or change the table in the field's row search ID.
—809	**Description:** RDSQL syntax error has occurred. **Solution:** Check that you have not misspelled an RDSQL statement, placed key words out of sequence, or included an *INFORMIX* SQL reserved word in your query.
—1202	**Description:** An attempt was made to divide by zero. **Solution:** Check that you are not attempting to divide a numerical column type by a character column type (for example. `16/Jones`) or that the value of the divisor does not equal zero.

Table 5-2: Phase 2 Historical Error Codes — *Continued*

—1203	**Description:** Values used in a MATCH must both be type CHARACTER. **Solution:** Check that the values included in your MATCH condition are both CHAR types. Use an alternate comparison condition for nonCHAR types.
—1204	**Description:** Invalid year in date. **Solution:** Acceptable years are 0001 to 9999. If two digits are used, RDSQL assumes the year is 19xx. Check the value entered in the date field.
—1205	**Description:** Invalid month in date. **Solution:** Months must be represented as the number of the month (1 through 12). Check the value entered in the date field.
—1206	**Description:** Invalid day in date. **Solution:** Days must be represented as the number of the day (01 through 31). Check the value entered in the date field.
—1226	**Description:** Decimal or money value exceeds maximum precision. **Solution:** Increase the precision of the DECIMAL or MONEY field.

Phase 2 Real-Time Error Codes

This section contains the Real-Time Database Manager error codes. Each code includes a description of the error and a recommended solution.

These errors are reported in the following format:

`CMS Database Manager error: <error number>`

In addition, a circumflex (^) appears in the listed Select statement(s) to mark the location of an error.

Table 5-3: Phase 2 Real-Time Error Codes

—1	**Description:** A syntax error has occurred. **Solution:** Check the select for misspelled keywords or keywords that are out of order.
—2	**Description:** An illegal character has been found in the select statement. **Solution:** Remove the illegal character (often a nonprintable control character).
—3	**Description:** The specified table name is invalid. **Solution:** Check the spelling of the table name and for required commas in the `From tables` field.
—4	**Description:** An invalid column has been specified (it is not found in any of the specified tables). **Solution:** Check the spelling of the column names.
—5	**Description:** A mixture of aggregates and nonaggregates are being selected, and this is not allowed in real-time reports. (This error code can also mean mismatched types in comparison.) **Solution:** Create two identical row search conditions, and apply one to the aggregate columns and one to the nonaggregate columns.
—6	**Description:** Bad column in the order by clause. **Solution:** Check that the column name in the order by clause is spelled correctly and that it is being selected by one of the fields included in this row search.

Table 5-3: Phase 2 Real-Time Error Codes — *Continued*

—7	**Description:** Bad index in order by clause. **Solution:** Order by clause has to have a column in the “select” clause or a number that indicates a position of the column in the “select” clause.
—8	**Description:** Bad argument given to an aggregate function. For example, you cannot take the SUM or AVG of a character column. **Solution:** Check the arguments for the aggregates and be sure that data type is appropriate.
—9	**Description:** In the “Select” of one of the fields associated with this row search, an action is being performed with the wrong data types. For example, you cannot use arithmetic with character fields. **Solution:** Check for these types of errors in the fields associated with the row search.
—10	**Description:** Error with subquery. **Solution:** There may be a subquery in the “where” clause that CMS does not support. Check the subqueries. This typically happens with an Agent Group report (a “select” embedded within a “select”).
—11	**Description:** CMS system error. **Solution:** Check the error logs.
—12	**Description:** Memory allocation error. **Solution:** Check the error logs.
—13	**Description:** Query cannot select more than one table. **Solution:** Check the error logs.

Running Custom Reports

You run custom reports using steps similar to steps you use to run standard reports. However, you run custom reports from the `Custom Reports` Main Menu option. Also, the Report Input window shows input fields you defined on the Define Input window for the report. For real-time reports, the Report Input window also automatically shows a `Refresh rate in seconds`

field. For historical reports, the Report Input window automatically shows report destination fields.

1. Select the `Custom Reports` Main Menu option. → *The Custom Reports submenu appears.*

2. Select either the `Real-time` or `Historical` submenu option. → *A submenu of custom reports, either real-time or historical, appears. The list includes all existing global reports and your private reports.*

3. Select the desired report. → *The Report Input window for the report appears.*

   Note
   If more than 20 custom reports are available, you may need to scroll the submenu to find the report you want. Scroll down with the down arrow (↓). Scroll up with the up arrow (↑).

4. Complete the fields of the Report Input window, and select `Run`. → `Working` *appears on the Report Input window status line. If the destination is the terminal, the report appears. If the destination is a printer or file,* `Successful` *appears.*

6 Advanced Report Design

Creating a Custom Data Table

The section of the *CentreVu*™ Call Management System (*CentreVu* CMS) database that stores historical ACD data uses the *INFORMIX* *SQL (ISQL) Relational Database Management System. All historical ACD data available for use in custom reports is stored in tables in the *CentreVu* CMS database. You can access *INFORMIX* and the *CentreVu* CMS database to build your own data tables to contain financial information, schedule information, product or service information, or any other type of information you want. You can then design historical custom reports to display the data, with or without ACD data.

Note

Instructions on the use of *INFORMIX* SQL appear in this document as a convenience. These instructions are not intended as a substitute for the *INFORMIX* documentation. Except where noted, the standard rules of *INFORMIX* SQL apply, as documented in the *INFORMIX-SQL Relational Database Management System User Guide* for *INFORMIX* SQL. If you purchased ISQL with the *CentreVu* CMS, this document is delivered with your *CentreVu* CMS software and documents.

Caution

CentreVu CMS does not automatically check the database for disk space used by data in custom tables. As a result, you can inadvertently fill up your disk with custom data. When this happens, you can lose or damage custom data and ACD data. Therefore, if you create custom data tables, be careful to check the amount of disk space available regularly. See the “System Setup” chapter in the *CentreVu*™ *CMS R3V5 Administration* (585-215-820) document for more information on disk storage.

* INFORMIX is a registered trademark of Informix Software, Inc.

Caution

If you back up data using the Maintenance — Backup Data window, the data stored in custom *INFORMIX* tables will be saved, but **the custom table definitions (table names, column names, data types, and so on) will not be saved**. As a result, if you lose the custom table definitions because of a disk crash, power hit, or some other reason, you cannot restore these table definitions using the Backup Data window. Because the custom table definitions will be lost, you will not be able to restore the custom data that was saved using the Backup Data window.

For this reason, it is recommended that you make your own backups of the custom table definitions using the *UNIX* [a] system/*Solaris* [b] system. If you lose *INFORMIX* table definitions and/or custom data, you can restore the table definitions using *UNIX* and then restore the custom data. See the "*UNIX* System/*Solaris* System" chapter in the *CentreVu* ™ *CMS Administration (*585-215-820) document for procedures on backing up data via the *UNIX* system.

a. UNIX is a registered trademark of Novell, Inc.
b. Solaris is a registered trademark of Sun, Inc.

To define custom data tables, use the steps described in the following pages.

Step 1: Access the *CentreVu* CMS Database in *INFORMIX*

To access the *CentreVu* CMS database in *INFORMIX*, use the following steps:

1a. Press the **Commands** SLK. → *The Commands submenu appears.*

1b. Select the `UNIX` option. → *All windows and menus disappear, and the **UNIX** prompt appears.*

1c. At the `$` prompt, type the following command: **`DBPATH=/cms/db/inf.`** Press the **Return** key. Type **`export DBPATH.`** Press the **Return** key. → *The **UNIX** prompt reappears.*

1d. At the `$` prompt, type the following pathname: **`/usr/informix/bin/isql.`** Press the **Return** key. → *The **INFORMIX** logo appears, followed by the **INFORMIX** Main Menu.*

```
INFORMIX-SQL:  Form Report Query-language User-menu Database Table Exit
 Run, Modify, Create, or Drop a form.
------------------------------------------- Press CTRL-W for Help --------
```

Step 2: Build the Table

In *INFORMIX*, you can select menu items in one of two ways:

- Use the arrow keys (→ or ←) to move the cursor to the menu option. Then, press **Return**.
- Type the first character of the menu option.

Note
To escape from a step and go back to the previous step, press **Delete** or **Del**, depending on your keyboard.

2a. At the *INFORMIX* Main Menu, select `Table`. → *The Select Database screen appears.*

```
SELECT DATABASE >>
Choose a database with the Arrow Keys, or enter name, then press Return.
------------------ cms ---------------------- Press CTRL-W for Help --------
CMS
```

2b. Type **`cms`**, and press **Return**. → *The Table menu appears.*

```
TABLE:    Create Alter Infor Drop Exit
Create a new table.
------------------ cms ---------------------- Press CTRL-W for Help --------
```

2c. Select `Create`. → *The Create Table screen appears.*

```
CREATE TABLE >>
Enter the table name you wish to create with the schema editor.
------------------ cms --------------------- Press CTRL-W for Help --------
```

2d. Type a name of up to 18 characters for the table you are building. The table name **must** begin with c_ (a lower-case "c" and underscore). For example, type **`c_workcode`**. Press the **Return** key.

→ *The Create Table menu appears with a highlighted box beneath "Column Name."*

```
CREATE TABLE c_workcode:        Add  Modify  Drop  Screen  Exit
Adds columns to the table above the line with the highlight.
------ Page 1 of 1 --------- cms -------- Press CTRL-W for Help --------
 Column Name                            Type        Length  Index  Nulls
 ..................................
 .                                  .
 ..................................
```

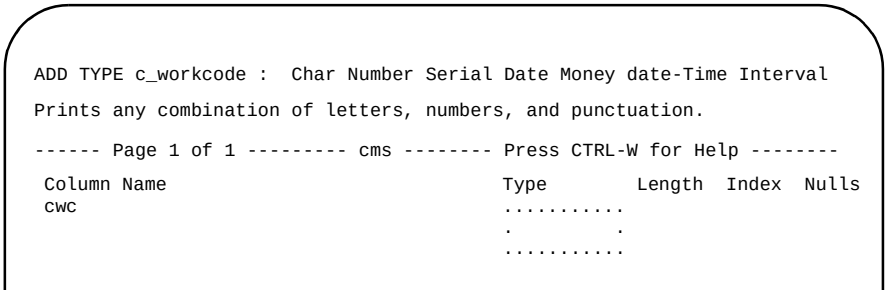
2e. Select Add.

→ *The Add Name screen appears.*

```
ADD NAME >>
Enter column name.  RETURN adds it.  INTERRUP returns to CREATE/ALTER menu.
------ Page 1 of 1 --------- cms -------- Press CTRL-W for Help --------
 Column Name                            Type        Length  Index  Nulls
 ..................................
 .                                  .
 ..................................
```

2f. Type a name of up to 18 characters for the column you are adding. If desired, you can use names of standard *CentreVu* CMS database items **only if** you enter the name in lowercase letters in the Dictionary subsystem. The Dictionary: Custom Items window accepts only lowercase letters for custom item names that are the same as standard database items. Press the **Return** key.

→ *The Add Type screen appears.*



The available data types are in the following table.

Table 6-1: Data Types

Char	Character (CHAR) columns store any combination of letters, numbers, and symbols.
Number	Numeric columns store numbers. The numbers stored can be one of the following five types: — **Decimal**. A decimal is a number that contains a decimal point. — **Smallint**. A Smallint column stores integers from -32,767 to +32,767. — **Integer**. An Integer column stores integers from -2,147,483,647 to +2,147,483,647. — **Smallfloat**. A Smallfloat column stores floating point numbers with up to seven significant digits. — **Float**. A Float column stores floating point numbers with up to 14 significant digits.
Serial	Serial columns store a unique sequence number in each row of the table.
Date	Date columns store calendar dates with the format **`mm/dd/yy`**.
date-Time	Not supported by CMS Custom Reports. However, it is supported by INFORMIX. You must type **T** in order to select date-Time. A date-Time column allows you to specify a *database* qualifier of year, month, day, hour, or minute.
Interval	Not supported by CMS Custom Reports. However, it is supported by INFORMIX. An interval column allows you to specify an *interval* qualifier of year, month, day, hour, or minute.
Money	Not supported by CMS Custom Reports. However, it is supported by INFORMIX. If your data will be dollars and cents, you should use **Decimal**.

2g. Select the type of data that the field will store. → *The additional prompts that appear differ depending on your selected data type.*

2h. Complete the definition of the column by responding to the additional prompts that appear. Which prompts appear and what order they appear in depend on the data type you selected. Table 6-2 lists the possible prompts. → *When you have responded to all of the prompts for the column, a new highlighted line appears in the table, and the* `Add Name` *field reappears at the top of the screen.*

2i. Repeat Steps 2f through 2h for each column you want to add.

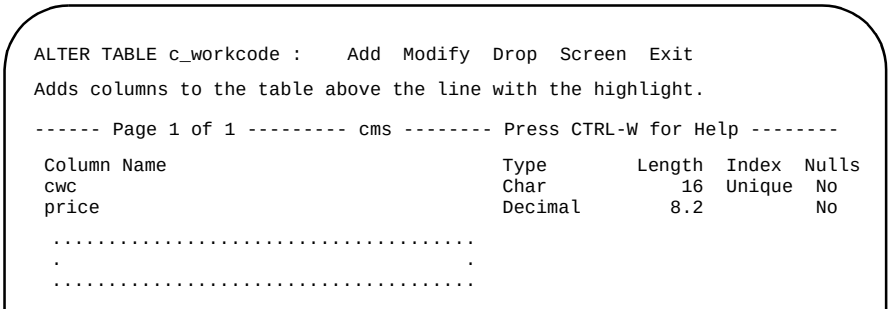
Table 6-2: Prompts for *INFORMIX* Data Types

Prompt	What to Do
`PRECISION`	Select either `Smallfloat` or `Float`. `PRECISION` appears for the `Float` type.
`INDEX`	Enter y if you want to make the column an index. `INDEX` appears for all types except `Serial`. A column with the `Serial` data type automatically becomes an index. Make the column an index only if the column will be used for row searches and the table will contain more than 200 rows of data.
`DUPLICATES`	Enter **y** if you want to allow the column to contain the same value in different rows. For example, if the column were to contain the last names of people, you would probably want to allow multiple entries (because, for example, you might have many different people with the last name of Smith). However, if you were creating a column of social security numbers, then you would enter **n** to prevent multiple entries of the same number. `DUPLICATES` appears for all types except `Serial`. **Note**: The `DUPLICATES` prompt displays if you answer yes to an `INDEX` prompt.

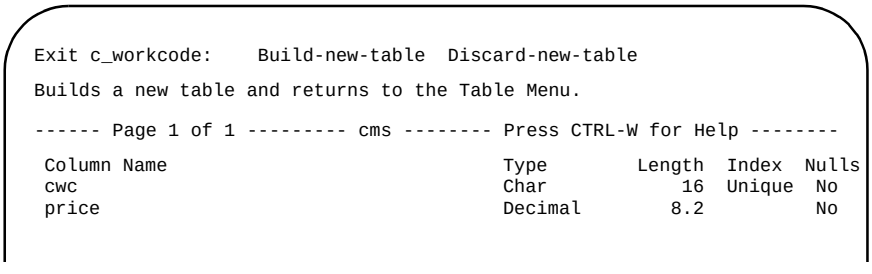
Table 6-2: Prompts for *INFORMIX* Data Types — *Continued*

Prompt	What to Do
NULLS	Enter **y** if you want to allow the column to have rows with no values (versus requiring values). For example, if your table is a list of customer data and the column you are adding is for the customer's employer, you might want to allow the NULL value for the case where the person is unemployed. Enter **n** if you want to require a value in each row. NULLS appears for all types except Serial.
ADD STARTING NUMBER	Enter the number that *INFORMIX* should use as a starting point for numbering rows. *INFORMIX* identifies the first row in the table with the number you enter. As each new row of data is added to the table, *INFORMIX* assigns that row the next number in the sequence. ADD STARTING NUMBER appears only for Serial.
NUMERIC	Type the first letter of the numeric you want: **I**nteger, **S**mallint, **D**ecimal, or **F**loat.
LENGTH	Enter the number of digits the column will store for a single piece of data. LENGTH appears for Char and Decimal types
SCALE	Enter the number of digits that should appear to the right of the decimal point. The decimal digits, but not the decimal point, occupy part of the field length you specified in the LENGTH field. SCALE appears after the LENGTH for Decimal.

2j. When you have added all columns, press the Return key until the Create Table screen appears. → *The Create Table menu appears.*



2k. Select `Exit`. → *The Build-new-table menu appears.*



2l. Select `Build-new-table`. → *The Table menu reappears. If no errors are found, then your table has been added. If errors are found, then you must go back to the Alter Table screen and correct these errors. You must then repeat steps 2k and 2l until the Table menu reappears.*

2m. Select `Exit` to return to the *INFORMIX* Main Menu. → *The **INFORMIX** Main Menu appears.*

Step 3: Add Data to the Table

To help you add data to the table, you must first create a data entry form associated with your table. For more information about forms, see “Creating Your Own Forms” in the *INFORMIX-SQL Relational Database Management System User Guide*. For more information about adding data, see “Entering Data,” in the same document.

3a. On the Main Menu, select FORM. → *The FORM menu appears.*

```
FORM:    Run  Modify  Generate  New  Compile  Drop  Exit
 Use a form to enter data or query a database.
----------------- cms ----------------- Press CTRL-W for Help --------
```

3b. Select GENERATE. → *The GENERATE FORM screen appears.*

```
GENERATE FORM >>
Enter the name you want to assign to the form, then press Return.
----------------- cms ----------------- Press CTRL-W for Help --------
```

3c. Enter a name (up to ten characters) for the form associated with your table. (If possible, use the same name as the table you created.) Press **Return**. → *The CHOOSE TABLE screen appears.*

```
CHOOSE TABLE >> c_workcode
Choose the table to be used in the default form.
----------------- cms ----------------- Press CTRL-W for Help --------
 workcode
```

3d. Enter the name of the table you want to enter data for, and press **Return**.

→ *The Table-selection-complete menu appears.*

```
GENERATE FORM:    Table-selection-complete   Select-more-tables   Exit
Continue creating a default form with the selected tables.
----------------- cms ----------------- Press CTRL-W for Help --------
```

3e. Select `Table-selection-complete`.

→ `The screen form specification was successfully compiled` *appears when processing of the form is complete, then the FORM menu appears.*

3f. Select `Run`.

→ *The RUN FORM window displays the table you selected.*

```
RUN FORM >>
Choose a form with Arrow keys, or enter a name, and press Return.
----------------- cms ----------------- Press CTRL-W for Help --------
 workcode
```

3g. Press the **Return** key.

→ *The PERFORM menu appears.*

```
PERFORM:  Query  Next  Previous  View  Add  Update  Remove  Table  Screen...
Searches the active database table.              ** 1: c_workcode table**
cwc                [                  ]
price              [         ]
```

3h. Select `Add`. → *The cursor moves to the first column in the table.*

3i. Enter data for the first column of the table, and press the Return key. → *The cursor moves to the next column.*

Note
If you get an error message, you may have entered data in the wrong format.

3j. Repeat Step 3i for each column.

3k. Press the Esc key to save the row of data. → `Row added` *appears. The row of data has been added to the table and saved.*

3l. Repeat Steps 3h through 3k for each row of data you want to add.

3m. Type **e** (for Exit) three times to exit *INFORMIX*. → *The UNIX prompt appears.*

3n. Press Ctrl B. → *The CentreVu CMS windows and menus that were displayed before you accessed UNIX reappear.*

Note
To design a custom report that uses data from the table, you **must** also enter the column names (database items) in the Dictionary: Database Item: Custom Items window.

Caution
CentreVu CMS does not automatically check the database for disk space used by data in custom tables. As a result, you can inadvertently fill up your disk with custom data. When this happens, you can lose or damage custom data and ACD data. Therefore, if you create custom data tables, be careful to regularly check the amount of disk space available. See the *CentreVu™ CMS R3V5 Administration* (585-215-820) document for more information on disk storage.

Modifying a Table

Use the following steps to add, change, or delete columns in an existing table.

1. On the *INFORMIX* Main Menu, select `Table`. → *The Select Database screen appears.*
2. Press the Return key. → *The TABLE menu appears.*
3. Select `Alter`. → *The ALTER TABLE screen appears. A list of existing tables also appears.*
4. Enter the name of the table you want to change, and press Return. → *The ALTER TABLE menu appears.*

Adding a Column

1. Select `Add` to add a new column. → *The ADD NAME screen appears.*
2. Complete the fields for the new column.
3. Press the Del key when you have finished adding the column(s), and go to Step 3 of "Deleting a Column." → *The ALTER TABLE menu appears.*

Changing a Column

1. Select `Modify` to change a column. → *The MODIFY NAME screen appears.*
2. Use the arrow keys to select a field to change.
3. Press the Del key when you have finished changing the column(s), and go to Step 3 of "Deleting a Column" in this section. → *The ALTER TABLE menu appears.*

Deleting a Column

1. To delete a column, use the arrow keys to move the cursor to the column. Select `Drop`. → *The REMOVE screen appears.*

2. Select `YES` to remove the column. → *The column disappears, and the ALTER TABLE menu appears.*
3. At the ALTER TABLE menu, select `Exit` when you are finished changing the table. → *The Build-new-table menu appears.*
4. Select `Build-new-table` to save your changes. Select `Discard-new-table` to ignore your changes.

Note

If you change columns in a table, then you may have to rebuild the form assigned to the table.

Changing Data in a Table

Use the following steps to add, change, or delete data in an existing table.

1. On the *INFORMIX* Main Menu, select `Form`. → *The FORM menu appears.*
2. Select `Run`. → *The RUN FORM screen and a list of forms appear.*
3. Enter the name of a form, and press Return. → *The PERFORM menu appears.*

Adding Rows of Data to a Table

1. Select `Add` to add rows of data. → *The table's column fields appear with the cursor in the first field.*
2. Enter data in the fields. Use Return or Tab to move between fields.
3. Press Esc when you have added a row of data. → *The PERFORM menu reappears. The message* `Row added` *also appears.*

Changing Rows of Data in a Table

1. To change data in a row, display the data you want to change.

   To do this, select `Query`.

   → *The table's column fields appear, with the cursor resting in the first field.*

2. Enter data in a column you want to search on, and press Esc.

   → *The column fields fill with data for that row.*

3. Select `Update`, and press the Return key.

   → *The Update screen appears.*

4. Use the arrow keys to move the cursor to the data you want to change. Overtype the data, and press Esc.

   → *The PERFORM menu reappears. The message* `This row has been changed` *also appears.*

Deleting Rows of Data From a Table

1. To delete a row of data, repeat Steps 1 and 2 of "Changing Rows of Data in a Table" in this section.

   Press Esc.

   → *The PERFORM menu reappears.*

2. Select `Remove`.

   → *The REMOVE ROW screen appears.*

3. Select `YES` to delete the row.

   → *The row of data disappears. The message* `Row deleted` *appears.*

Including Forecast Data in a Custom Report

Note

Forecasting is a separately purchased feature of *CentreVu* CMS. If you have not purchased Forecasting, you cannot run forecasts and, therefore, cannot include forecast data in a custom report.

For historical custom reports only, you can design reports that include current day forecast data. Only current day forecast data is available for custom reports because it is the only forecast data saved in the *CentreVu* CMS database.

The steps for creating a custom report with forecast data are almost identical to the steps for creating any other custom report. As with any other type of data, you must specify the database item(s), the table(s), the row search conditions, and the report input fields.

However, there are two differences:

1. You **cannot** copy a standard current day forecast report on the Screen Painter. The Current Day Forecast does not appear if you select `List all` in the Get Copy of Design window.
2. *CentreVu* CMS stores current day forecast data in two separate tables:

`f_cday`	This table primarily contains forecast administrative data entered in the Current Day Configuration window.
`f_cdayrep`	This table contains the agent positions required and forecast calls carried data, as well as objectives entered in the Call Handling Profiles window.

See the *CentreVu CMS R3V8 Database Items and Calculations* (585-210-939) for a complete list of the database items these tables contain. See the *CentreVu™ CMS R3V5 Forecast* (585-215-825) document for more information about Forecasting.

If you were to design a very abbreviated version of the current day forecast report, the design might look something like that shown in Figure 6-1.

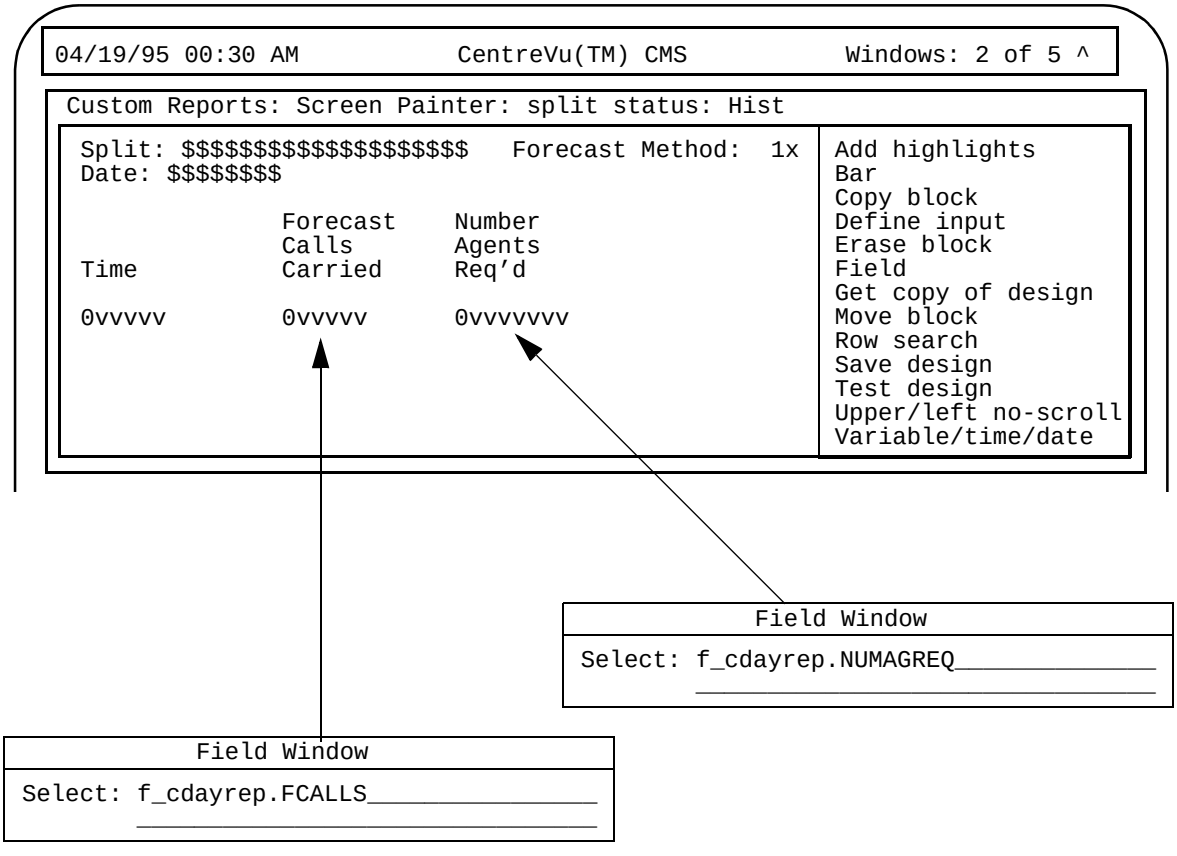


Figure 6-1: Sample Forecast Report Design

In the report in Figure 6-1, Row Search ID 0 is assigned to the `Time`, `Forecast Calls Carried`, and `Number Agent Req'd` fields. The conditions for Row Search ID 0 might be as follows:

```
Row Search Window

Row search ID: 0
From table(s): f_cdayrep
Select rows where: ROW_DATE = $i_date and SPLIT = $i_split
                   and STARTTIME = $i_time and ACD = $acd
Field/bar type (Select one):
 < > Discrete
 <x> Repeated vertically, spacing 1___
```

Figure 6-2: Row Search Conditions for Forecast Data — Sample 1

Notice that, as in historical interval reports for splits, the statement in Row Search ID 0 searches for rows based on values for STARTTIME, ROW_DATE, and SPLIT.

In the report in Figure 6-1, Row Search ID 1 is assigned to the `Forecast Method` field. Notice also that the database item for this field is FMETHOD. FMETHOD is stored in the `f_cday` table, but is **not** stored in the `f_cdayrep` table. This is an example of how current day forecast data is divided between the two tables.

However, notice that Row Search ID 1 (Figure 6-3) has row search conditions that are almost identical those in Figure 6-2:

```
                    Row Search Window

Row search ID: 1
From table(s): f_cday
Select rows where: ROW_DATE = $i_date and SPLIT = $i_split
                   and ACD = $acd
Field/bar type (Select one):
 <x> Discrete
 < > Repeated vertically, spacing 1___
```

Figure 6-3: Row Search Conditions for Forecast Data — Sample 2

The only differences between Row Search ID 0 and Row Search ID 1 are the table selections and the field types. In this way, the data in the tables will match.

Note

In the standard Current Day Forecast, **FMETHOD** normally displays a character string, either `Seasonal trending`, `Current trending`, or `No trending`. However, the *CentreVu* CMS database actually stores a number to represent each method. So, if you include the **FMETHOD** database item in a report, *CentreVu* CMS displays a number, not a character string, in the report (see the *CentreVu CMS R3V8 Database Items and Calculations* document (585-210-939) for a description of these numbers).

Note

You can retrieve data from the Current Day Configuration Forecast (`f_cday`) table only if the appropriate call handling profile has been specified for the particular dates. For retrieval of data from the Current Day Forecast Report (`f_cdayrep`) table, the Forecast Manager must have run for the particular date for which the report is run.

Including Exceptions Data in a Custom Report

For historical custom reports only, you can design reports that include exceptions data. See the *CentreVu CMS R3V8 Database Items and Calculations* (585-210-939) document for a description of the exceptions tables and the data they contain.

The steps for creating a custom report with exceptions data are almost identical to the steps for creating any other custom report. As with any other type of data, you must specify the database item(s), the table(s), the row search conditions, and the report input fields. However, you cannot copy any standard exceptions report on the Screen Painter. The exceptions reports do not appear if you select `List all` in the Get Copy of Design window.

Also, in every exceptions table, the database item EXTYPE stores the types of exceptions that occurred. However, exception types are stored as numbers, not as character strings. If you want your report to list the types of exceptions that occurred, then the types must be listed as numbers (see the *CentreVu CMS R3V8 Database Items and Calculations* (585-210-939) document for a description of these numbers). The VDN Exceptions Report shown in Figure 6-4 is an example of this.

```
VDN Exceptions
ACD: burbank                          Printed 04/10/93 04:35 PM
Date: 04/09/93

                                                                        Time
Time      VDN            Vector         Exception                      Limit   Threshold
-------------------------------------------------------------------------------------------
09:16AM   Catalog Sales  Weekday Sales  Number calls abandoned in vector       20
09:24AM   Catalog Sales  Weekday Sales  Calls in ACD split queue               10
10:02PM   Catalog Sales  Off-hour Sales Calls disconnected                    50
```

Figure 6-4: VDN Exceptions Report — Sample 1

If you designed a custom exception report to replicate the standard VDN Exceptions Report, then the design might appear as shown in Figure 6-5.

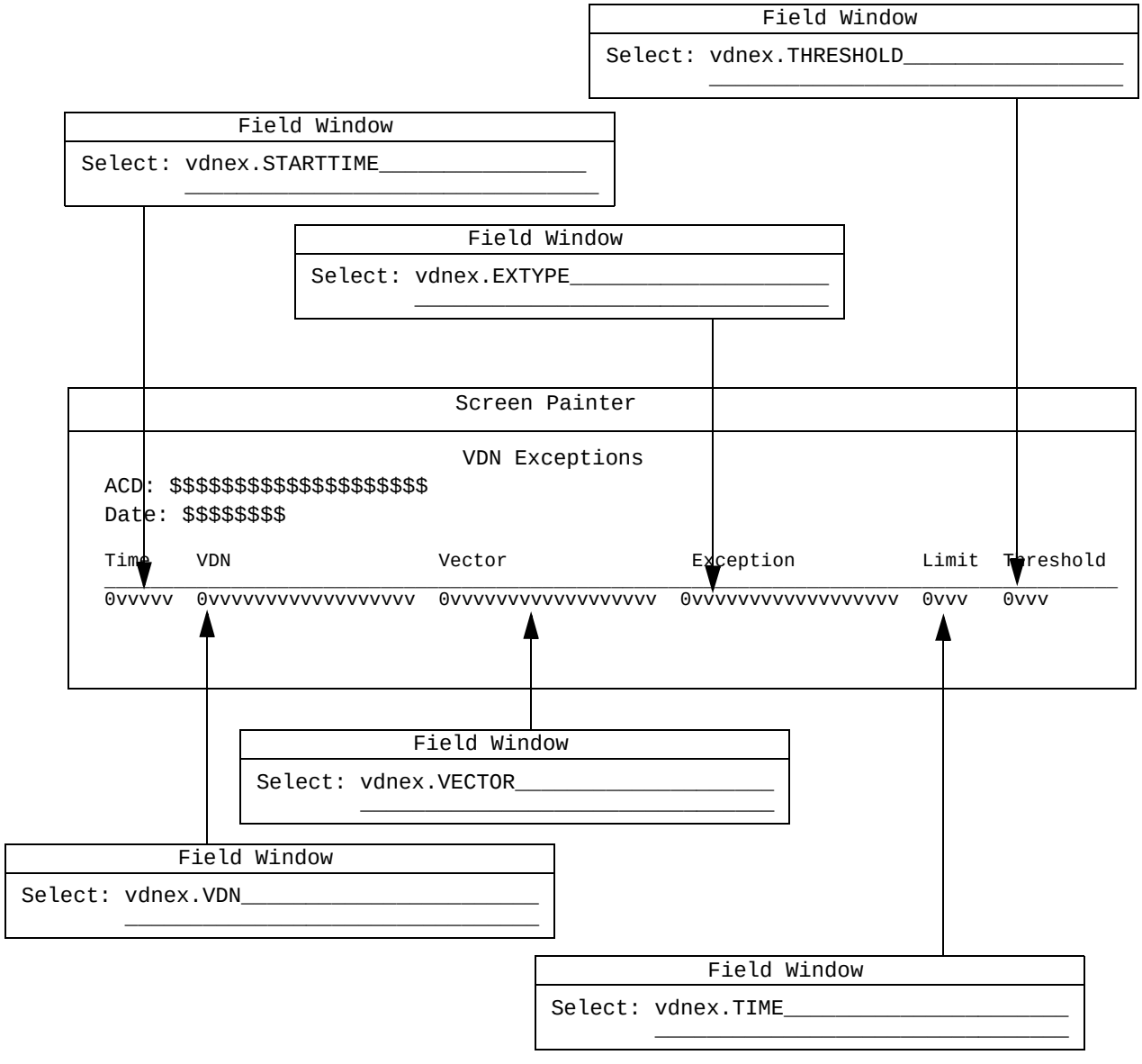


Figure 6-5: Custom Exceptions Report — Sample 1

Figure 6-6 (shown on the previous page) is an example of the report that results from this design shown in Figure 6-5.

```
VDN Exceptions
ACD: burbank                          Printed 04/10/93 04:35 PM
Date: 04/09/93

                                                                                Time
Time      VDN            Vector         Exception                             Limit   Threshold
-------------------------------------------------------------------------------------------------
09:16AM   Catalog Sales  Weekday Sales  37                                    32      20
09:24AM   Catalog Sales  Weekday Sales  37                                    30      10
10:02PM   Catalog Sales  Off-hour Sales 76                                    73      50
```

Figure 6-6: VDN Exceptions Report — Sample 2

The report would appear this way because *CentreVu* CMS exceptions tables store the exception types as numbers.

However, another way of designing an exception report would be to use **`count(*)`** for the exception field. You could then include a specific exception type as part of the row search (Figure 6-7). *CentreVu* CMS would then count the rows that had that exception type and display the total in the report.

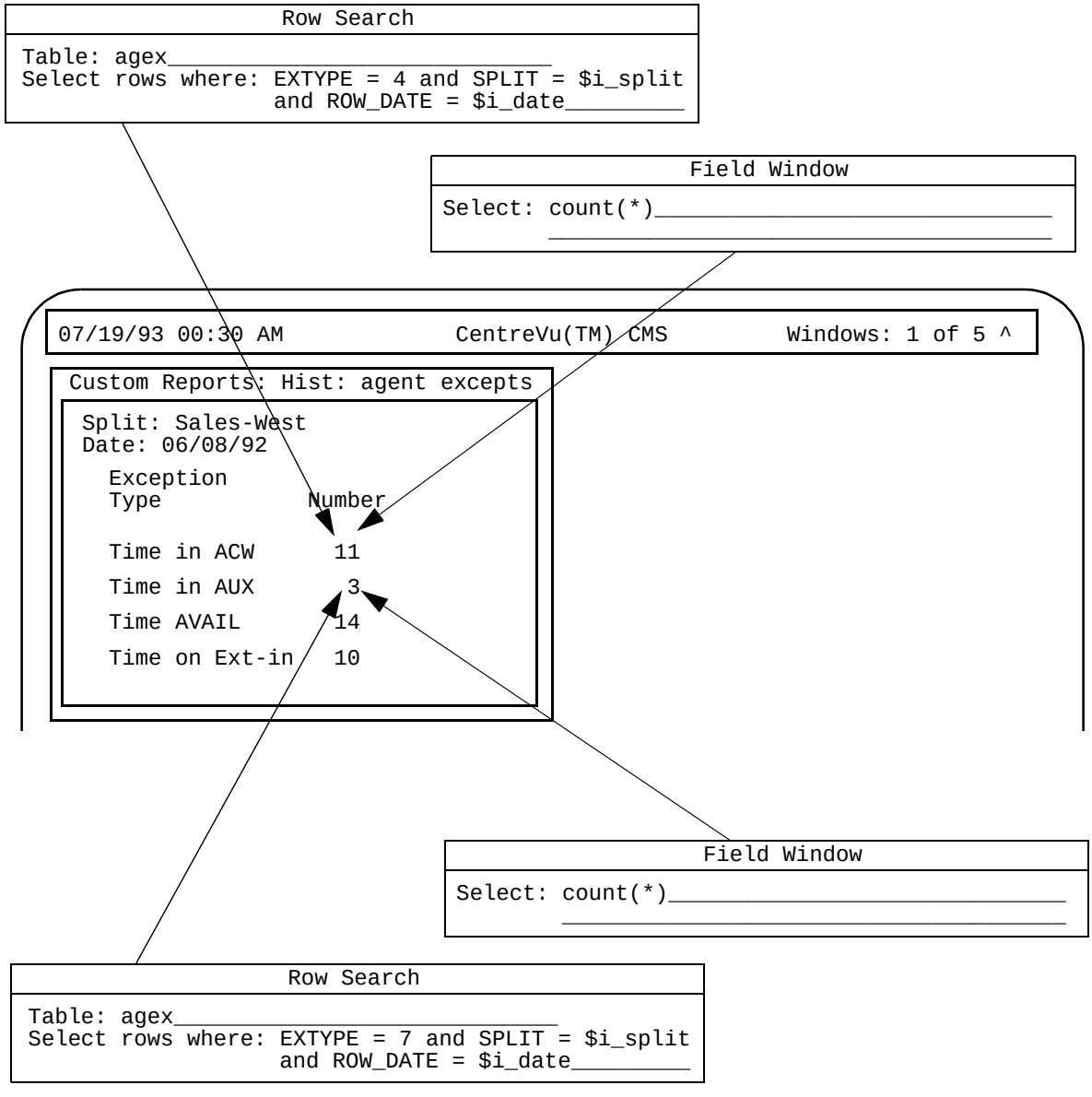


Figure 6-7: Custom Exceptions Report — Sample 2

In the example, the row search statement for the first field searches for exception type #4 (Time in ACW) for a particular date and split. Because the field is **`count(*)`**, the field displays the total number of rows that had exception type #4 for the date and split.

Similarly, the row search statement for the second field searches for exception type #7 (Time in AUX) for a particular date and split. Again, because the field is **`count(*)`**, the field displays the total number of rows that had exception type #7 for the date and split.

Note

For retrieval of data from an exceptions table, you can only retrieve data for those exceptions that have been turned on for the particular split, agent, etc., have actually occurred and have not been deleted from the database because they exceeded the storage parameters for exceptions.

Selecting Rows from More Than One Table

For historical reports only, you can merge data from two tables into a single report field. You can, for example, take the number of ACD calls a single agent handled (where data is taken from the `dagent` table) and divide by the total ACD calls handled by the agent's split (where data is taken from the `dsplit` table). Or, you can take the ACD calls a split handled in an intrahour interval (where data is taken from the `hsplit` table) and divide by the total ACD calls the split handled for the day (where data is taken from the `dsplit` table).

To merge data from two tables into a single report field:

- The two tables **must** have at least one database item in common. Typically, the database items in common are indexes.
- Both table names must be entered in the `Table` field of the Row Search ID assigned to the field.
- For custom historical reports — you can use data from more than one table and use the same row search ID for multiple tables.
- At least one "join" clause must appear in the Row Search ID assigned to the field. A "**join" clause** searches for the same value for the same item in both tables. In this way, the data extracted from the rows in both tables will be related.

  A "join" clause has the following format:

  `tablename1.item1 = tablename2.item1`

  Where `item1` is a database item that the tables have in common.

- You **must** prepend the table name to each database item included in the `Select rows where:` statement. This rule applies to all database items, even those that are not in a "join" clause.
- A "join" clause must use a database item that is also included with a regular “where” clause (one that directly assigns a value or variable name to the database item).

Note
The syntax of a "join" clause is similar to that in standard *INFORMIX* SQL syntax. See the *INFORMIX-SQL Relational Database Management System User Guide* for *INFORMIX* SQL.

Caution
If you run a report that merges data from two tables (particularly tables with large amounts of data) into a single field and your `Select rows where:` statement is not specific enough, you may get an error message when you test the design. The specific cause may be that the number of selected rows is very large, and *CentreVu* CMS does not have enough space to create temporary files. If this is the case, you should add additional “where” clauses to the row search criteria.

Often, you need several “join” clauses in a Row Search criteria statement. For example, you must first specify row search values for either the `hagent` or `hsplit` table if you define a report field that contains the following data expression:

```
hagent.ACDCALLS/hsplit.ACDCALLS
```

You might enter the following “where” clauses:

```
hsplit.SPLIT = $i_split and hsplit.ROW_DATE = $i_date and
hsplit.STARTTIME = $i_time and hsplit.ACD = $acd
```

Note
Notice that table names are prepended to each database item.

You must then specify "join" clauses so that the rows found in one table are related to the values found in the other table. In this example, you would add "join" clauses (shown in bold) for every regular “where” clause:

```
hsplit.SPLIT = $i_split and hsplit.ROW_DATE = $i_date
and hsplit.STARTTIME = $i_time and hsplit.ACD = $acd
and hsplit.SPLIT = hagent.SPLIT and hsplit.ROW_DATE =
hagent.ROW_DATE and hsplit.STARTTIME =
hagent.STARTTIME and hsplit.ACD = hagent.ACD
```

The report design shown in Figure 6-8 illustrates how this row search selection would affect data in a report.

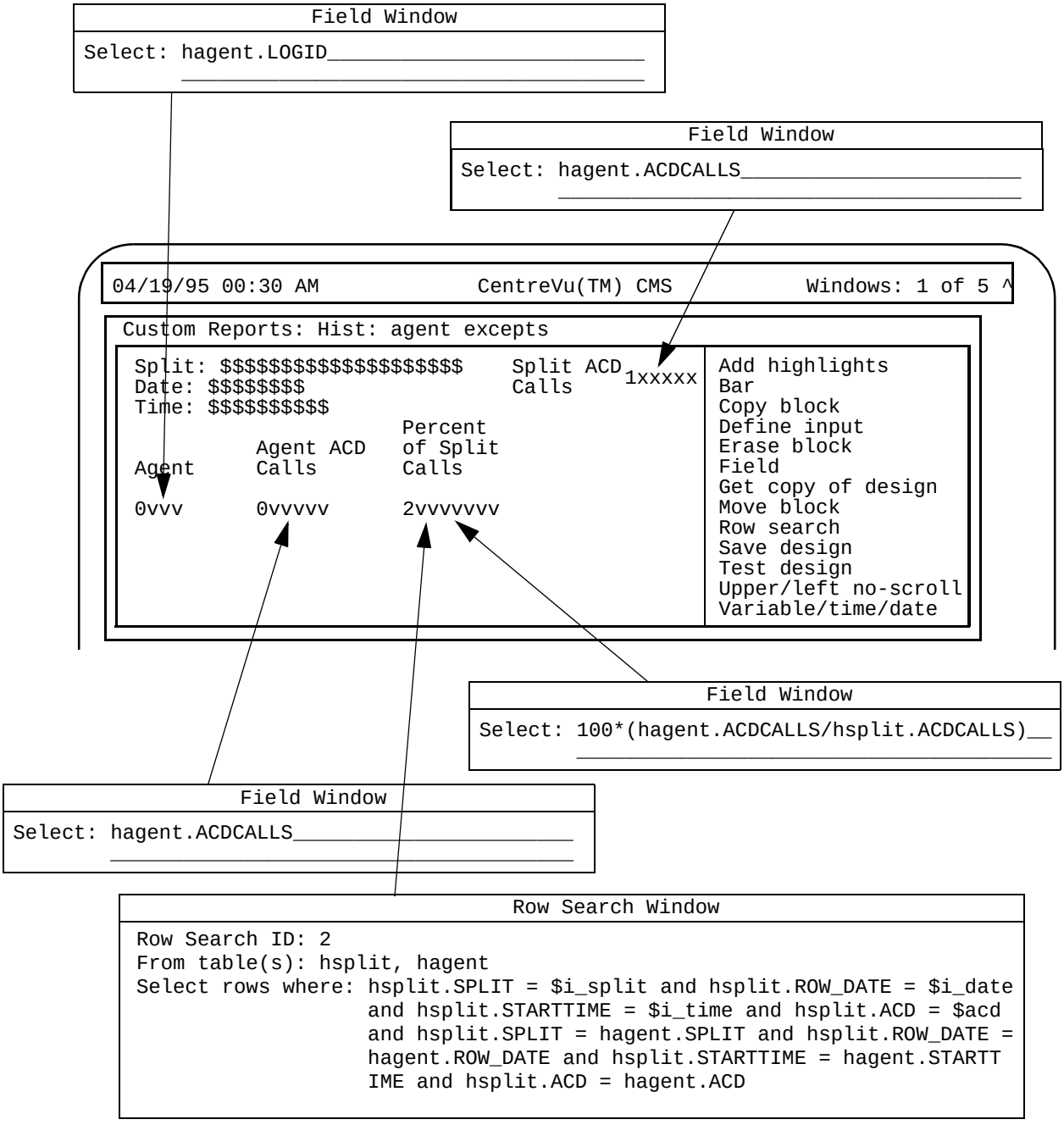


Figure 6-8: Report Design with Data from Two Tables Merged in a Field

Notice that the `Percent of Split Calls` field merges data as discussed in our example. Also notice that Row Search ID #2, which is assigned to the `Percent of Split Calls` field, contains the row search statement as discussed in our example.

To illustrate the effect of "join" clauses, the report design in Figure 6-8 includes the `Agent ACD Calls` and `Split ACD Calls` fields. Notice that each field uses a database item that is also included in the `Percent of Split Calls` field.

When the report is run, the `Percent of Split Calls` field, for each agent in the selected split, divides that agent's ACD calls by the total ACD calls for the split, then multiplies by 100 to give a percentage. Thus, as in Figure 6-9, if agent 1000 had 23 ACD calls, and the split Sales-West had 412 ACD calls, the `Percent of Split Calls` for agent 1000 would be `5.583` (100 * [23/412]).

```
04/19/95 00:30 AM        CentreVu(TM) CMS       Windows: 1 of 5

Custom Reports: Hist: split status

  Split: Sales-West                Split ACD
  Date: 06/08/93                   Calls
  Time: 8:00am

                                   Percent
                 Agent ACD         of Split
  Agent          Calls             Calls
   1000          23                5.583
   1004          18                4.369
   1005          18                4.369
   1009          21                5.097
   1010          24                5.825
   1012          23                5.583
```

Figure 6-9: Sample Report with Merged Data

Alternative Row Search Conditions

For your row search conditions, you will most often use the “where” clauses described in Chapter 4. However, several additional “where” clause formats are available. This section describes those formats.

Selecting Rows Based on a Range of Values

If you want a range of splits included in the report, you might use two “where” clauses as in the following example:

```
Select rows where: SPLIT >= 1 and SPLIT <= 5
```

Then, when you run the report, *CentreVu* CMS finds rows for Splits 1 through 5, as shown in Figure 6-10.

SPLIT	ACDCALLS	ABANDONS	ACDTIME	ABNTIME	
1	443	48	36898	988	. . .
2	234	37	20012	777	
3	111	20	13111	400	. . .
4	652	59	53442	1058	
5	451	32	27635	644	. . .
6	93	11	15321	245	
7	509	43	35401	851	. . .
8	391	31	19768	603	
9	142	10	9786	203	. . .
10	480	39	33389	789	

Figure 6-10: Sample Use of Ranges in Where Clauses

Note

In most cases, the easiest way to specify a range is to define a report input field that accepts a range. See the “Selecting Rows Based on a Range of Values” section in this chapter.

Using Apostrophes for Some Database Item Values

Some database items require that, if you **hardcode** values in a “where” clause, you enclose the values in apostrophes ('), as shown in the following example.

```
Select rows where: ROW_DATE > ’07/01/93’
```

Standard database items whose values you must enclose in apostrophes are:

VDN (the value is a VDN number)
EXTENSION (the value is an extension number)
LOGID (the value is an agent login ID)
EQLOC (the value is a 9-digit trunk location number)
ROW_DATE (the value is a date in mm/dd/yy format)
CWC (the value is a call work code of 1 to 16 digits)

Also, any custom database items you define as CHAR or DATE columns in *INFORMIX* are items whose hardcoded values you must enclose in apostrophes.

For historical reports only, an alternative format for these database items is:

```
Expression matches ’value’
```

In this type of clause, **matches** is the same as **=**. However, this type of clause lets you use wildcard searches. That is, within the apostrophes, you can use an asterisk (*****) or a question mark (**?**).

The asterisk (*****) matches any and all characters, including blanks and no characters. Look at the following examples:

```
Select rows where: EQLOC matches ’01*’
```

This clause finds all rows where the EQLOC value begins with 01, which would mean all trunks for module 01. Thus, 010012020, 010211110, and 011023100 are values that would match.

```
Select rows where: ROW_DATE matches ’*/01*’
```

This clause finds all rows where the ROW_DATE value has 01 as its dd (day of the month). That is, the clause searches for the first day of each month. Thus, 01/01/93, 04/01/93, and 10/01/92 are values that would match.

Note

Actually, ***/01*** would also find all dates, if they had passed, in the year 2001 (for example, 01/22/01, 08/03/01, and 11/31/01).

The question mark (**?**) matches any single character. Look at the following examples:

```
Select rows where: EXTENSION matches '444?'
```

This clause finds all rows where the EXTENSION value is four digits and has 444 as the first three characters. Thus, 4441, 4440, and 4449 would match. However, 444 would not match.

```
Select rows where: LOGID matches '?000'
```

This clause finds all rows where the LOGID value is four digits, begins with any number, and ends with "000." Thus, 4000, 5000, and 9000 would match. 000 would not match.

You can use **not** in a "matches" clause to exclude rows. For example, you may want to exclude a range of login IDs from your report with a clause like the following:

```
Select rows where: LOGID not matches '2*'
```

This clause finds all login IDs except those starting with "2." If you have 4-digit login IDs, login IDs from 2000 to 2999 would be excluded.

Using String-Value Database Items

String-value database items contain numerical data that the Dictionary subsystem translates to display current states or state changes. For example, a report field using the string-value database item WORKMODE displays AVAIL, ACD, ACW, and so on, depending on what state the particular agent is currently in. However, tables store states as numbers, not strings. When you run a report, *CentreVu* CMS simply substitutes the character strings for the numerical values. This process is identical to the substitution of names for split numbers, trunk group numbers, vectors, and so on.

If you want to use string-value database items in a "where" clause, then you must specify numerical values, not string values.

For example, if you want a current real-time agent report that lists data only for agents on extension-in or extension-out calls, then you can select the Current Interval Agent table and enter a statement as follows:

```
Select rows where: SPLIT = $splitvar and WORKMODE >
10 and WORKMODE < 60
```

This statement finds rows for a user-specified split where the agent state is one of the following:

ACWIN (numerical value of 20)
ACWOUT (numerical value of 30)
AUXIN (numerical value of 40)
AUXOUT (numerical value of 50)

See the *CentreVu CMS R3V8 Database Items and Calculations* (585-210-939) document for a complete list of row search values for string-value database items.

Other Available Formats for "Where" Clauses

For historical reports only, you can specify a list of hardcoded values using the following format:

```
Expression in (list of values)
```

As with a basic "where" clause, the `Expression` can be a database item or calculation. The values you list in the parentheses must be separated by commas. Also, you must use apostrophes for those values that normally require them. For example, the following "where" clause specifies three dates for the report:

```
Select rows where: ROW_DATE in ('07/01/93',
'07/08/93','07/15/93')
```

The following "where" clause specifies three splits for the report:

```
Select rows where: SPLIT in (1,7,22)
```

For historical reports only, you can also exclude rows using a list of values by adding **`not`** to the clause. Using **`not`** can be extremely useful if you want to exclude, for example, lunch time intrahour intervals from your intrahour historical reports, as shown in the following example:

```
Select rows where: STARTTIME not in (1130, 1200,
1230).
```

Repeating Aggregate Function Values

Normally, a field (or bar) that contains an aggregate function (**sum, max, min,** or **avg**) displays just one value. This is true no matter what your "Select rows where" criteria are. Therefore, you would normally select "discrete" as the field/bar type in the row search ID assigned to aggregate functions.

However, you may at times want *CentreVu* CMS to display multiple values for an aggregate function. **In historical reports only**, *CentreVu* CMS can display multiple values for an aggregate function. For *CentreVu* CMS to do this, the aggregate function **must** share a row search ID with one or more fields that do not have an aggregate function.

For example, you may want a report (Figure 6-11) that shows the maximum (max) ACD calls in an interval for each of a variety of splits.

```
                Date: 07/01/93

                     Max ACD Calls-
Split                Morning
1                    652
2                    491
3                    297
```

Figure 6-11: Repeated Aggregate Function Report

This report would have the following design.

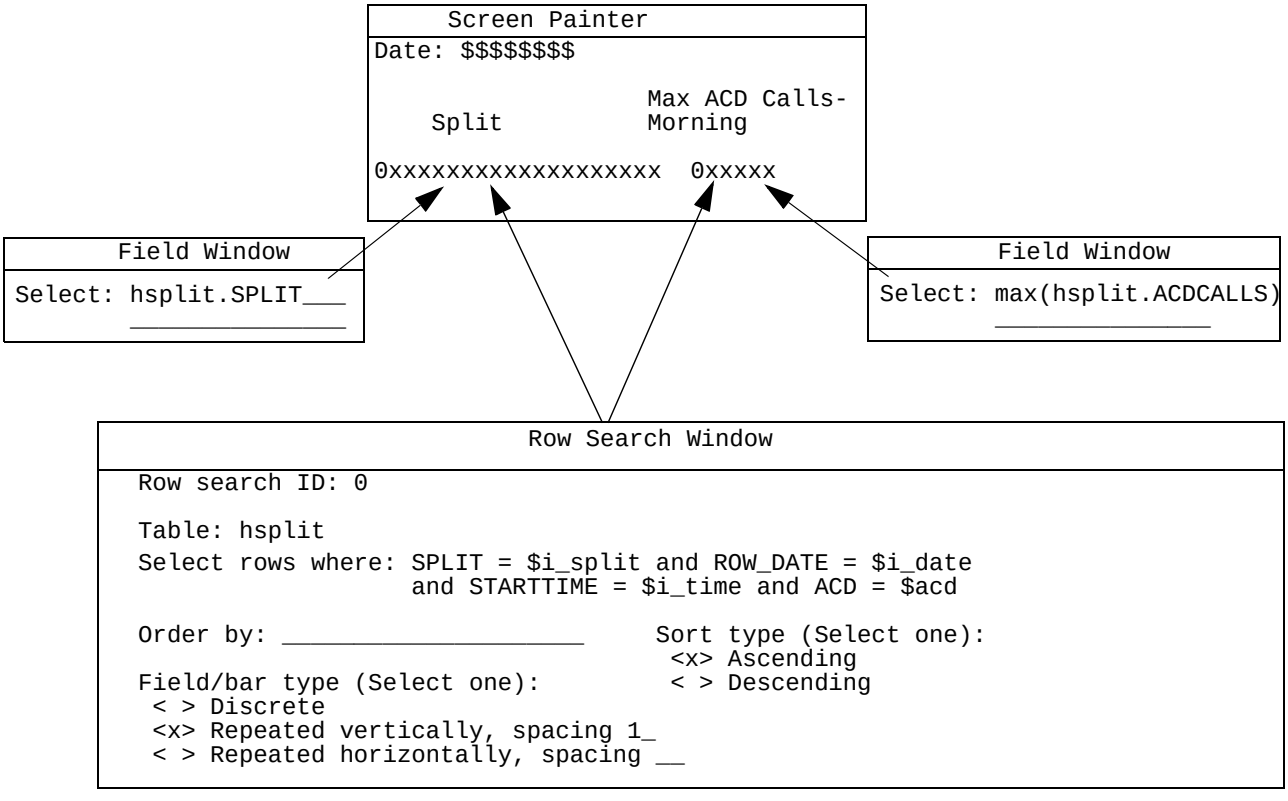
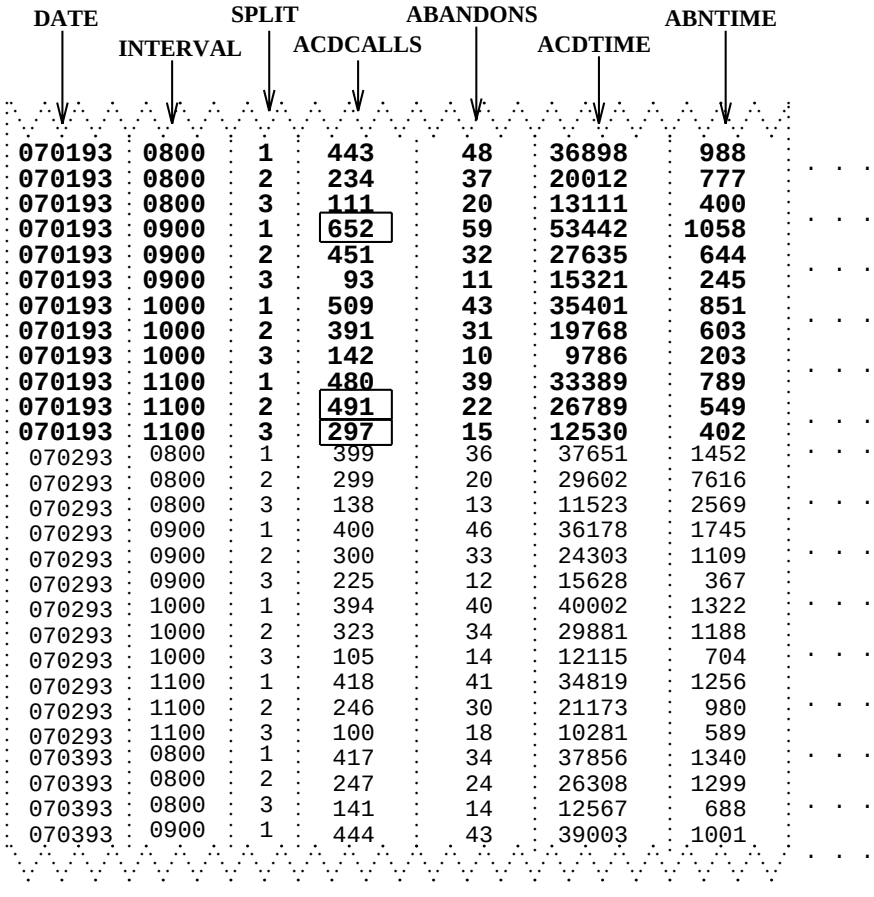


Figure 6-12: Sample Report Design for Repeated Aggregate Functions

According to the design, *CentreVu* CMS searches the Intrahour Split table (Figure 6-13) for the splits the user ordered when running the report (Splits 1, 2, and 3 in our example). *CentreVu* CMS also searches for date (07/01/93 in our example) and the maximum ACD calls (in boxes) in the selected range of intrahour intervals for each of the selected splits.



DATE	INTERVAL	SPLIT	ACDCALLS	ABANDONS	ACDTIME	ABNTIME
070193	0800	1	443	48	36898	988
070193	0800	2	234	37	20012	777
070193	0800	3	111	20	13111	400
070193	0900	1	652	59	53442	1058
070193	0900	2	451	32	27635	644
070193	0900	3	93	11	15321	245
070193	1000	1	509	43	35401	851
070193	1000	2	391	31	19768	603
070193	1000	3	142	10	9786	203
070193	1100	1	480	39	33389	789
070193	1100	2	491	22	26789	549
070193	1100	3	297	15	12530	402
070293	0800	1	399	36	37651	1452
070293	0800	2	299	20	29602	7616
070293	0800	3	138	13	11523	2569
070293	0900	1	400	46	36178	1745
070293	0900	2	300	33	24303	1109
070293	0900	3	225	12	15628	367
070293	1000	1	394	40	40002	1322
070293	1000	2	323	34	29881	1188
070293	1000	3	105	14	12115	704
070293	1100	1	418	41	34819	1256
070293	1100	2	246	30	21173	980
070293	1100	3	100	18	10281	589
070393	0800	1	417	34	37856	1340
070393	0800	2	247	24	26308	1299
070393	0800	3	141	14	12567	688
070393	0900	1	444	43	39003	1001

Figure 6-13: Sample Row Search of Grouped Aggregate Functions

CentreVu CMS would then display the values found in the report. However, notice in Figure 6-13 that *CentreVu* CMS found four rows with the SPLIT value of 1, four rows with SPLIT value 2, and four rows with SPLIT value 3, but only displayed each value once in the report. This is because the aggregate function `max(hsplit.ACDCALLS)` field shares the same row search ID with the `hsplit.SPLIT` field.

If the `hsplit.SPLIT` field and the `max(hsplit.ACDCALLS)` fields were assigned different row search IDs (with each ID having identical criteria), the report for Splits 1, 2, and 3 would show data as shown in Figure 6-14:

```
      Date: 07/01/93

            Max ACD Calls-
Split       Morning
1           652
1
1
1
2
2
2
2
3
3
3
3
```

Figure 6-14: Sample Display of Group Values

The display of a value only once to represent multiple occurrences of the same value is called **grouping** and can be done only when sum, max, min, or avg values are listed based on the unique values found for other fields. The rules for grouping data are:

- If you assign a row search ID to one or more aggregate function fields, you may assign the same row search ID to a maximum of eight nonaggregate fields.
- Assigning the same row search ID to both aggregate functions and nonaggregate fields makes sense only if the nonaggregate fields contain identifier data (split numbers, login IDs, dates, interval start times, vector numbers, and so on).
- If you assign the same row search ID to both aggregate functions and a nonaggregate field, the report displays a single value for each unique value found for the nonaggregate field (that also matches the row search criteria). If multiple rows contain the same value, then *CentreVu* CMS still lists the value once. For this reason, only identifier fields should have the same row search ID as aggregate functions.
- If you assign the same row search ID to aggregate functions and **more than one** nonaggregate field, then the report displays a single row of data for each unique **combination** of values for the nonaggregate fields.

A Database Items and Calculations

General Information

Reference

Please refer to the *CentreVu® CMS R3V8 Database Items and Calculations* (585-210-939) document, for full descriptions of the database items and calculations avaialable through the *CentreVu* Call Management System (CMS).

Glossary

Access permissions Permissions assigned to a CMS user so that user can access different subsystems in CMS or administer specific elements (splits/skills, trunks, vectors, etc.) of the ACD. Access permissions are specified as **read** or **write** permission. Read permission means the CMS user can access and view data (for example, run reports or view the Dictionary subsystem). Write permission means the CMS user can add, modify, or delete data and execute processes.

Administrator, CMS A CMS user who has permissions for most or all parts of CMS. In Custom Reports, a CMS Administrator can access the designs of all custom reports, even those of which the administrator is not the owner.

Aggregate function A prefix (avg, max, min, or sum) attached to a database item, calculation, parts of a calculation, or a calculation name. An aggregate function normally displays a single value that is determined from a group of selected values.

Ascending Listed with the lowest values first and the highest values last. With time and dates, the oldest values are listed first.

Associated ACD The ACD associated (or assigned to) an input field, as defined in the Define Input window. When an input field is associated with an ACD, the values the user enters into the field will apply to that ACD only. For example, if an input field requiring a split number also has associated ACD #1, the selected split number will be a split in ACD #1.

Bar A representation of data in the form of a bar that gets longer or shorter as values go up or down. A horizontal bar grows and shrinks horizontally. A vertical bar grows and shrinks vertically.

Block A rectangular area on the Screen Painter that you define and use to quickly rearrange report fields, bars, and text.

Block editing Defining and using a block to copy, move, or delete fields, bars, and text in a report design.

Calculation A calculation is also a formula consisting of database items, numbers, and arithmetic operators (`+, -, /, *,` and `( )`). A calculation can also include constants.

Calculation name A name for a calculation, either a standard CMS calculation name or a name that you define in the Dictionary subsystem. A calculation name can be useful if you need to use the same calculation for field/bar definitions in multiple custom reports. Then, if you need to change the calculation, you can change it once in the Dictionary, and all custom reports that use the calculation name will reflect the change.

Call-based items
The category of database items in CMS that are committed to the database after the call completes. If a call starts and ends in different intrahour intervals, all of the call-based data is recorded in the interval in which the call completed. Most database items are call-based.

Call Management System (CMS)
A software product used by business customers that have Lucent Technologies telecommunications switches/ECS and receive a large volume of telephone calls that are processed through the Automatic Call Distribution (ACD) feature of the switch/ECS. The CMS collects call-traffic data, formats management reports, and provides an administrative interface to the ACD feature in the switch/ECS.

CMSQL
A tool that allows direct queries of the historical database. This tool is the interactive interface typically used to view the *INFORMIX* database. For CMS purposes, CMSQL is used instead of *INFORMIX* SQL.

Column
A column is a part of a table that stores a particular type of data (a number of events of a certain type, a length of time spent on a certain type of event, the time an event happened, a numerical identifier of an ACD entity, or the current status of an ACD entity). Database items are the names of columns. Examples are ACDCALLS, ACDTIME, SPLIT, STARTTIME, WORKMODE, and so on.

Constant
A name you assign in Dictionary to a fixed numerical value. A constant can be useful if you need to use the same numerical value for field/bar definitions in multiple custom reports. Then if you need to change the value, you can change it once in Dictionary, and all custom reports that use the constant will reflect the change.

Count(*)
An expression you can enter in the `Select` field in the Field or Bar window. `count(*)` tells CMS to count the number of rows in a table that match certain row search conditions (as defined for the assigned Row Search ID) and display the total in the field/bar.

Custom database item
A database item (column name) which you have included in a custom INFORMIX table and which you have identified to CMS in Dictionary.

Custom report
A report that you create and design using the Custom Reports subsystem.

Data expression
An expression you enter in the `Select` field of the Field or Bar window to define the data and manipulation of that data for display in a field or bar.

Database
A group of files that store ACD data according to a specific time frame: current and previous intrahour real-time data and intrahour, daily, weekly, and monthly historical data.

Database item

A name for a specific type of data stored in one of the CMS databases. A database item may store ACD identifiers (split numbers or names, login IDs, VDNs, etc.) or statistical data on ACD performance (number of ACD calls, wait time for calls in queue, current states of individual agents, etc.). Examples are ACDCALLS, ACDTIME, SPLIT, WORKMODE, and so on.

Database tables

CMS uses these tables to collect, store, and retrieve ACD data. Standard CMS items (database items) are names of columns in the CMS database tables.

Date format

The standard format for entering dates on CMS reports. Acceptable formats are:

- Month/day/year (for example, 3/21/96)
- A "-" offset based on today's date (for example, -1 for yesterday)
- Separating individual data entry items using a semicolon (for example, 3/21/96;3/23/96;3/25/96)
- Entering ranges by placing a hyphen between entries (for example, 3/21/96-3/25/96).

When you specify a date for a weekly report, that date must correspond to the week start day selected in the System Setup—Storage Intervals window. If the date and day do not match, the message *No records found* displays in the status line.

The month start date must be the first day of the month.

Define Input Window

The secondary window you access on the Screen Painter to define fields for the report's Report Input window.

Descending

Listed with the highest values first and the lowest values last. With time and dates, the most recent values

Design

The physical layout of a custom report, as well as the definition of the report's fields, bars, input window, and row search criteria.

Direction (Bar)

The direction in which a bar grows and shrinks, either horizontally or vertically.

Discrete field/bar

A field or bar for which a single value will be found and displayed.

Dictionary

A CMS subsystem that can be used to assign names to various call center elements such as login IDs, splits/skills, trunk groups, VDNs and vectors. These names appear on reports, making them easier to interpret. Dictionary also allows customized calculations to be created for use in reports.

Entity A generic term that refers to one of the following: Agent, Split/Skill, Trunk, Trunk Group, VDN, or Vector.

Field A space designated in a custom report to display a specifically-defined piece of ACD data. Data is displayed in a field as characters in specific format (that is, as time with am/pm, as decimal, as date, and so on).

Global access The ability for other CMS users to run a custom report and to copy its design on the Screen Painter.

Historical database A database that contains intrahour records for up to 62 days in the past, daily records for up to 5 years in the past, and weekly/monthly records for up to 10 years in the past for each CMS-measured agent, split/skill, trunk, trunk group, vector, and VDN.

Historical reports Reports that display past ACD data for various agent, split/skill, trunk, trunk group, vector, or VDN activities.

INFORMIX A relational database management system used to organize CMS historical data.

***INFORMIX* SQL** A query language tool that is used to extract data from an *INFORMIX** database. For the CMS historical database, CMSQL is used in place of *INFORMIX* SQL.

Index A column (database item) that causes the values in a row to be related. For example, in the Current Interval Split table, `SPLIT` is an index. An index adds structure to the data in a table.

Input field An input field is an area on a user window into which a *CentreVu* CMS user enters one or more valid field values. For example, the valid values for the input field `Split` are integers from 1 through 255 (Generic 3r Version 2 to Version 4) and split names that have been assigned in the Dictionary subsystem.

Input window The window that appears when you are running a report so you can specify what data [what split/skill, what agent(s), what times, what dates(s), and so on] the report will display.

* INFORMIX is a registered trademark of Informix Software, Inc.

Interval-based items A category of database items. These items generally represent the amount of time during a collection interval spent doing a particular activity. Interval-based items are updated throughout the collection interval and timing is restarted at the end of the interval. Interval-based items should only be used to show the amount of time in an interval for an activity or to calculate percentages of time spent in an interval. Interval-based items should not be used to calculate averages (such as average hold time).

Intrahour interval A 15-, 30-, or 60-minute segment of time starting on the hour. An intrahour interval is the basic unit of CMS report time.

Justification The lining up of data for a field such that in the report, the data is always centered, lined up on the left, or lined up on the right.

Maximum graph value The value that a bar will represent when it is at its maximum length or height.

Measured A term that means an ACD element (agent, split/skill, trunk, trunk group, vector, VDN) has been identified to the switch as of interest to the CMS. The switch sends messages to CMS only for ACD elements that are measured. If the ACD element is not measured on the switch, no data is collected.

Name (Synonym) fields Fields in which you may enter a name (synonym) that has been entered in the Dictionary subsystem (for example, names of agents, splits/skills, agent groups, trunk groups, vectors, VDNs).

Normal user A user that normally has access to a limited number of subsystems within CMS. A normal user **cannot** access a custom report design created by another user and can neither run nor copy the private reports of another user.

Owner The CMS user who created the design of a custom report.

Pattern matching Searching the database for data that partly or totally matches a set of characters, (letters, numbers, and symbols) entered by a user. For example, the user may enter `*01*`, and CMS might find the following types of matching data: `0001, split01, 22010,` or `01444.`

Private access The restriction that only the owner of a custom report can run the report. Also, no other normal user can copy the report's design. Private access does **not** restrict CMS administrators from running a private report or accessing its design.

Private report A custom that only the creator can access.

Read permission	The CMS user with read permission can access and view data (for example, run reports or view the Dictionary subsystem). Read permission is granted from the User Permissions subsystem.
Real-time database	Consists of the current and previous intrahour data on each CMS-measured agent, split/skill, trunk, trunk group, vector, and VDN.
Real-time reports	Reports that display current ACD call activity on agents, splits/skills, trunks, trunk groups, vectors, and VDNs for the current or previous intrahour interval. Current intrahour interval real-time reports are periodically updated as data changes during the interval. Previous intrahour interval real-time reports show data totals for activity that occurred in the previous intrahour interval.
Repeated horizontally	The specification that multiple values will be found for a field/bar and that the values will be displayed in a vertical series of fields or bars.
Repeated vertically	The specification that multiple values will be found for a field/bar and that the values will be displayed in a horizontal series of fields or bars.
Report type	The specification of a custom report as either a real-time or historical report.
Reversed thresholds	The reversal of thresholds so that the bar has the color of normal conditions when it is at its longest. With reversed thresholds, the bar will change from normal to caution to warning as the bar shrinks in length.
Row	A single horizontal line of data in a table. The data is related by the value(s) of one or more columns. For example, each row of current real-time agent data contains data for a specific agent login ID.
Row search ID	The identification number of a set of row search conditions defined in the Row Search window. This number is assigned to fields/bars that will use the associated set of row search conditions. Therefore, the available row search IDs 0 to 9.
Row search window	The secondary window accessed on the Screen Painter that you use to define row search criteria and assign row search criteria to field/bars.
Scale	A line, with or without tick marks, that appears in a custom report to provide a reference point for the approximate value of a bar as it changes length.
Screen Painter	The window that you use to design custom reports. The Screen Painter has special properties that make its operation different from that of other windows.

Second threshold A graph term for the upper limit you enter for a particular condition in a graph report. When this limit is met, the bar(s) change color/intensity, indicating that a possible Warning condition exists.

Secondary window A user window that is generated from a primary window. Secondary windows can be moved, sized, or scrolled and do not count in the user window count.

Select The name of the first field in the Field and Bars windows. You enter the data expression you want for a report field/bar in the `Select` field.

Select rows where statement The row search criteria CMS uses to retrieve data from the database. `Select rows where` statements typically consists of several "where" clauses.

Sort order The order in which you want data to appear in a custom report. The sort order is always based on the values of one or more database items. Real-time reports allow the use of only one database item for sorting.

Sort type The order, ascending or descending, in which data should be displayed.

Standard database item A database item (column name) for a column in a standard CMS table. Standard database items are listed in the Dictionary subsystem, and a description of all database items are in the *CentreVu CMS R3V8 Database Items and Calculations* (585-210-939) document.

String (Data format) A report field that displays data from a CHAR column. CHAR columns, which store data as characters, include `LOGID, VDN, CWC,` and so on.

String (Input field) An input field type that allows users to type character strings and pattern searches when running the report. Any input field that has the "string" field type must be associated in the Row Search window with a database item or table column that is a CHAR column. This includes `LOGID, VDN, CWC,` and other database items.

Structured Query Language (SQL) A language used to interrogate and process data in a relational database (such as *INFORMIX*). See *INFORMIX SQL*.

Subsystem Each CMS main menu selection (for example, Reports, Dictionary, System Setup, Exceptions, etc.) along with Timetable and Shortcut is referred to as a subsystem of the Call Management System.

Table An array of columns and rows that stores data for a type of ACD element (split/skill, agent, trunk, and so on) and for a specific time frame (for the current intrahour interval, for past intrahour intervals, for past days, and so on).

Text The words, numbers, and other characters you type directly onto the Screen Painter to provide report titles, column headers, row identifiers, and other labels.

Threshold A value at or above which CMS indicates a change in condition, either to a caution (first threshold) or a warning (second threshold) condition. Thresholds apply to bars in a report.

Tick marks The marks on a scale defined for a bar in a report. Tick marks break a scale into segments to indicate increments of the bar's length relative to the maximum graph value.

Type (Input field) The specification of how CMS is to validate a user's input values and format when ordering the report. That is, you can tell CMS to check for numbers, character strings (names), times, dates, specific ACD configuration values (split numbers, login IDs, and so on), etc.

Type (Report) The specification of whether a report is a real-time or historical report. You specify the type on the Report Select window.

Time Format The standard format for entering times on CMS reports. Acceptable formats are:

- A 12-hour time format with AM/PM (for example, 7:30AM, 5:00PM).
- A 24-hour time format (for example, 7:30, 17:00).
- A "-" offset based on the current interval date (for example, -1 for the previous hour or -0:30 for the previous half hour interval).

User ID The login ID for a CMS user. CMS links custom report designs to a user's user ID. That is, the owner of a report design is the user (identified by user ID) who created the design.

User Permissions A CMS subsystem that allows the CMS administrator to define user access permissions.

Variable name The name assigned to a report input field that will allow the user to enter values and have those values used to find data for the report. The variable name links the row search criteria to the report input fields so that the user's input values become the basis of the search for data.

Where clause An expression in the `Select rows where` field that specifies values for a single database item or calculation. A "where" clause has the format `Expression` **`Relational operator`** `Value.` A row search criteria statement can consist or multiple "where" clauses.

Index

Symbols

"Where" Clauses
Basic . . . 4-42
Exclusions . . . 4-45
Multiple . . . 4-44
Other Formats for . . . 6-32
Range/List Variable . . . 4-45
Selecting Rows from an ACD . . . 4-45
Using String-value Items . . . 6-31
Using to Select a Range of Values . . . 6-29
Variable . . . 4-44
$ Not Defined . . . 5-10

Numerics

1 . . . 5-19
11 . . . 5-20
12 . . . 5-20
13 . . . 5-20
2 . . . 5-19
3 . . . 5-19
4 . . . 5-19
5 . . . 5-19
6 . . . 5-19
8 . . . 5-20
9 . . . 5-20

A

Access . . . 2-2, 2-5, 2-8
Defining . . . 2-5
Screen Painter . . . 3-1
Access the Report Select Window . . . 2-2
ACD
Variable Name . . . 4-8
Agent Trace Data . . . 2-6
agex . . . 4-41
Aggregate Functions . . . 4-18
And/Or, Using in Clauses . . . 4-44
Apostrophes, Using for Database Items . . . 6-30
Arithmetic Operators . . . 4-16
Arrow Keys . . . 3-2
Assign a Name . . . 2-3
Associated ACD . . . 4-8

B

Back . . . 3-2
Back Space . . . 3-2
Background Color
Default . . . 5-3
Specific Color . . . 5-3
Backing Up to Save Custom Reports . . . 6-2
Bars . . . 4-27
Changing a Definition . . . 4-36
Data Expressions for . . . 4-30
Defining . . . 4-27, 4-30, 4-34, 4-35
Direction of . . . 4-30
Discrete . . . 4-46
First Threshold . . . 4-34
Maximum Graph Value . . . 4-33
Position and Length . . . 4-28
Repeated . . . 4-46
Repeated Horizontally . . . 4-46
Repeated Vertically . . . 4-46
Saving a Definition . . . 4-36
Scales for . . . 4-31
Second Threshold . . . 4-34
Table for Calculation Names . . . 4-30
Bars Reversing Threshold Colors . . . 4-35
Basic Tasks of Designing a Custom Report . . . 1-4
Blinking . . . 5-2
Block
Defining . . . 3-8
Block Editing . . . 3-8
Copying . . . 3-13
Erasing . . . 3-9
Moving . . . 3-10
Block, definition . . . 3-8
Bright . . . 5-2
Busy Hour report . . . 1-6

C

cagent . . . 4-40
Calculation Names . . . 4-16
Defining Tables for . . . 4-21, 4-30
Calculations . . . 4-16, A-1
Call Work Code Data . . . 2-6
call_rec . . . 4-41
Cannot Mix Aggregates and Nonaggregates in the Select . . . 5-10
Cannot Use the SYN Function for Order By . . . 5-11
ccwc . . . 4-40
Change
Access . . . 2-8
Description . . . 2-8
Changing
a Bar Definition . . . 4-36
a Field Definition . . . 4-27
No-Scroll Area . . . 5-6
Reports Name, Access, and Type . . . 2-8
Changing a Report's Access or Description . . . 2-8
Changing Report Input Fields . . . 4-11
Changing Row Search Conditions . . . 4-55
CHAR Colums
Definition . . . 6-6
Checklist of Design Tasks . . . 1-4

Clauses 4-42
CMS Documents P-8
CMS System Error
  Check the Error Log 5-11
  Data Collection Off 5-11
  Too Much Data Retrieved 5-11
  Updating Translations 5-12
Column
  Changing 6-14
  Deleting 6-14
Column Names, Custom Tables
  Adding 6-6
Column Types 6-6
Columns in a Table
  Adding 6-14
  Changing 6-14
  Deleting 6-14
Constants 4-15
Conventions Used in Document P-5
Copy a Report 3-7
Copying a Block 3-13
Copying an Existing Report 3-4
count(*) 4-20
csplit 4-40
ctkgrp 4-40
Ctrl-b 3-2
Ctrl-d 3-2
Ctrl-e 3-4
Ctrl-f 3-2
Ctrl-u 3-2
Ctrl-x 3-4
Ctrl-y 3-4
Ctrl-z 3-4
ctrunk 4-40
Current ACD 4-8, 4-57, 4-59
Current Interval P-6
Current Interval, Definition P-6
Custom Database Items 4-15
Custom Report
  Definition 1-2
Custom Tables 6-1
  Adding Data 6-16
  Changing a Column 6-14
  Changing Data 6-17
  Deleting a Column 6-14
  Deleting Data 6-17
cvdn 4-40
cvector 4-40

D

dagent 4-40
Daily Data P-6
Daily Data, Definition P-6
Data
  Adding Data in a Custom Table 6-16
  Changing Data in a Custom Table 6-17
  Deleting Data in a Custom Table 6-17
  Merging 1-3
  Ordered by Date 4-48, 4-49
  Ordered by Split 4-49
  Ordered by Split and ACD Calls 4-50
Data Expressions 4-14
  Aggregate Functions 4-18
  Calculation Names 4-16
  Calculations 4-16
  Constants 4-15
  count(*) 4-20
  Data from More Than One Table 4-20
  for Bars 4-30
Data in a Table
  Adding 6-11, 6-16
  Changing 6-16
  Deleting 6-16
Data Retrieval 1-11
  How CMS Retrieves Data 1-11
Data Storage 1-7
  How CMS Stores Data 1-7
Data Types 6-8
Database
  How CMS Retrieves Data 1-11
  How CMS Stores Data 1-7
  What is a Database Item 1-10
  What is a Database Table 1-7
Database Item
  What is It 1-10
Database Items A-1
  Custom 4-15
  Not in Standard Reports 1-7
  Standard 4-14
Database Table Names 4-40
Date 4-57, 4-60
Date Formats 4-22, 4-58
Date Report Was Run 4-58
dcwc 4-41
Default 5-2, 5-3
Default Value for Input Field 4-5
Define a Report 2-2
Define Access to Report 2-5
Define as Real-Time or Historical 2-6
Define Input Window 4-1
Define User ID 2-5
Defining
  a Report 2-2
  Access 2-5
  Aggregate Functions 4-18
  An Associated ACD 4-8
  Bar Data Expression 4-30
  Bar Direction 4-30
  Bar Position and Length 4-28
  Bar Scale 4-31

Defining, (continued)
Bars . . . 4-27
Block . . . 3-8
Calculation Names . . . 4-16
Constants . . . 4-15
Data Expression . . . 4-14
Data Expression for Fields . . . 4-14
Data for a Custom Report . . . 4-1
Default Value for Input Field . . . 4-5
Display Format . . . 4-57
Display Type . . . 4-57
Field Format . . . 4-22
Field Length . . . 4-13
Field Position . . . 4-13
Fields . . . 4-11
First Threshold for a Bar . . . 4-34
Input Field Type . . . 4-5
Input Fields . . . 4-1
Maximum Graph Value . . . 4-33
No-Scroll Areas . . . 5-4
Number of Columns . . . 4-4
Number of Rows . . . 4-4
Order of Input Fields . . . 4-10
Report Type . . . 2-6
Reports as Real-time or Historical . . . 2-6
Row Search ID . . . 4-39, 4-51
Rows . . . 4-37, 4-42, 4-45, 4-46, 4-48, 4-49, 6-30, 6-31
Second Threshold for a Bar . . . 4-34
Single Value or Range/List . . . 4-4
Sort Order For DataStep 8
Select a Sort Order . . . 4-49
Sort Type . . . 4-48
Tables for Calculation Names . . . 4-21, 4-30
User ID . . . 2-5
Variable Names . . . 4-2
Defining a Report's Name, Access, and Type . . . 2-2
Defining Normal/Reverse Threshold . . . 4-35
Defining Report Fields . . . 4-11
Delete . . . 2-8
Delete a Report . . . 2-8
Deleting a Custom Report . . . 2-8, 2-9
Description . . . 2-4, 2-8
Description, Entering a Report . . . 2-4
Design Options . . . 1-2
Designing Reports
Basic Tasks . . . 1-4
Dim . . . 5-2
Discrete . . . 4-46
Disk Space . . . 6-1, 6-13
Display Options . . . 4-58, 5-1
Document
Conventions . . . P-5
Organization . . . P-2
Other . . . P-8
Use . . . P-2
Down . . . 3-2
dsplit . . . 4-40
dtkgrp . . . 4-41
dtrunk . . . 4-41
dvdn . . . 4-41
dvector . . . 4-41

E

Editing Keys . . . 3-2
Editing Keys on the Screen Painter . . . 3-4
Editing With Blocks . . . 3-8
Enter a Report Name . . . 3-6
Enter a User ID . . . 3-7
Entering Text . . . 3-13
Erase All Fields . . . 3-4
Erase Characters . . . 3-4
Erase Field . . . 3-4
Erasing a Block . . . 3-9
Error Codes
1 . . . 5-19
11 . . . 5-20
12 . . . 5-20
13 . . . 5-20
2 . . . 5-19
3 . . . 5-19
4 . . . 5-19
5 . . . 5-19
6 . . . 5-19
8 . . . 5-20
9 . . . 5-20
Historical . . . 5-14
Real-Time . . . 5-19
Error Messages . . . 5-10
$ Not Defined . . . 5-10
Cannot Mix Aggregates and Nonaggregates in the Select . . . 5-10
Cannot Use the SYN Function for Order By . . . 5-11
CMS System Error
Check the Error Log . . . 5-11
Data Collection Off . . . 5-11
Too Much Data Retrieved . . . 5-11
Updating Translations . . . 5-12
Historical . . . 5-14
INFORMIX-SQL . . . 5-14
Keyword AVG Invalid in Where Clause . . . 5-13
Keyword BETWEEN Invalid for Real-Time . . . 5-13
Keyword COUNT Invalid in Where Clause . . . 5-13
Keyword MAX Invalid in Where Clause . . . 5-13
Keyword MIN Invalid in Where Clause . . . 5-13
Keyword SUM Invalid in Where Clause . . . 5-13
Keyword SYN Invalid in Where Clause . . . 5-13
Real-Time . . . 5-19
Test Design . . . 5-10
Exception Reports . . . P-6
Exceptions Data . . . 2-6
Including in Report . . . 6-21
Excluding Rows of Data . . . 4-45
Existing Reports, Copying . . . 3-4

F

f_cday 4-41
f_cdayrep 4-41
Field
  Data Format 4-11
  Format 4-11
  h 4-12
  Length 4-11
  Position 4-11
  v 4-12
  x 4-12
Field Prompt 4-4
Field Type 4-5
Field Window 4-13
Field With Data from More Than One Table . . . 6-25
Fields
  Assigning Row Search ID 4-51
  Changing Definition 4-27
  Changing Row Search ID 4-56
  Data Expressions for 4-14
  Date Formats 4-22
  Defining . 4-11, 4-14, 4-15, 4-16, 4-18, 4-19, 4-20, 4-21, 4-22, 4-23, 4-25, 4-27
  Defining Data Expression 4-14
  Discrete 4-46
  Formats for 4-22
  Highlighting 5-1
  Justifying Data in 4-21
  Point-In-Time Formats 4-26
  Position and Length 4-13
  Repeated 4-46
  Saving Definition 4-27
  Showing Run Time/Date and User Inputs . . . 4-56
  Synonym Types 4-24
  Time Formats 4-25
Forecast Data 2-6
  Including in Report 6-18
Format of Fields 4-11
Formats
  Date 4-58
  Time 4-58
Formats for Fields 4-22
Forward 3-2
From Tables 4-39
Functions, Aggregate 4-18

G

Global Access 2-5
Graphs 4-27

H

H 4-28
h 4-12
hagent 4-40
Hardcoded Values 4-44
hcwc 4-41
Highlighting
  Fields 5-1
  Options 5-2
Highlighting Options 5-2
  Blinking 5-2
  Bright 5-2
  Default 5-2
  Dim 5-2
  Normal 5-2
  Reverse Video 5-3
  Underline 5-3
Historical 2-6
Historical Database P-6
Historical Database, Definition P-6
Historical Reports P-6
Historical Reports, Definition P-6
Historical, Defining Report as 2-6
How CentreVu CMS Retrieves Data 1-11
How CentreVu CMS Stores Data 1-7
hsplit 4-40
htkgrp 4-40
htrunk 4-41
hvdn 4-41
hvector 4-41

I

Index
  What is It? 1-10
INFORMIX
  Accessing 6-3
  Adding Data to the Table 6-11
  Building Tables 6-4
  Column Types 6-6
  Columns in a Table, Adding 6-14
  Columns in a Table, Changing 6-14
  Columns in a Table, Deleting 6-14
  Creating Tables 6-1
  Data in a Table, Adding 6-16
  Data in a Table, Changing 6-16
  Data in a Table, Deleting 6-16
  Data Types 6-8
  Documentation 6-1
Input Field Type,Defining 4-5
Input Fields 4-1
  Accessing Define Input Window 4-2
  Associating an ACD with the Variable Field . . 4-8
  Changing 4-11
  Columns 4-4
  Default Value 4-5
  Defining 4-1, 4-2, 4-8, 4-9
  Order of 4-10
  Pattern Matching 4-7
  Prompt 4-4
  Rows 4-4

Input Fields, (continued)
Saving a Variable Field 4-9
Single Value or Range/List 4-4
Variable Names for 4-2
Input Fields That Allow Pattern Matching 4-7
Input Variable 4-59
Input Variables 4-38
Insert 3-4

J

Justifying Data 4-21

K

Key Combinations 3-2
Keyword AVG Invalid in Where Clause 5-13
Keyword BETWEEN Invalid for Real-Time 5-13
Keyword COUNT Invalid in Where Clause 5-13
Keyword MAX Invalid in Where Clause 5-13
Keyword MIN Invalid in Where Clause 5-13
Keyword SUM Invalid in Where Clause 5-13
Keyword SYN Invalid in Where Clause 5-13

L

Length of Fields 4-11
Lenth 4-13
linkex 4-41
List
Variables 4-38

M

magent 4-40
Matches Operator 6-30, 6-31
mctex 4-41
mcwc 4-41
Merging Data 1-3
Monthly Data P-6
Monthly Data, Definition P-6
Movement Keys on the Screen Painter 3-2
Moving a Block 3-10
Moving Cursor on Screen Painter 3-2
msplit 4-40
mtkgrp 4-41
mtrunk 4-41
Multi-ACD 4-9
Multi-ACD by Split/Skill report 1-6
Multi-ACD report 1-6
Multiple Clauses 4-44
Multiuser Mode P-6
mvdn 4-41
mvector 4-41

N

Name 2-2, 2-3, 3-6
Non-customizable Reports 1-6
Normal 5-2
No-Scroll Areas 5-4
Changing 5-6
Not Matches Operator 6-31
Number of Columns
Defining 4-4
Number of Rows
Defining 4-4

O

One Value 4-46
Open Screen Painter 3-1
Operators
And/Or 4-44
Arithmetic 4-16
Matches 6-30, 6-31
Not Matches 6-31
Relational 4-42
Order
for Input Window Prompts 4-10
for Sorting 4-49
Organization of Document P-2
Overview of CMS Database 1-7

P

pagent 4-40
Pattern Matching 4-7
pcwc 4-40
Permissions 2-5
Phase 1 Error Messages 5-10
Point-In-Time Formats 4-26
Position 4-13
Position of Fields 4-11
Preface
Organization of This Document P-1
R3V2 CMS Publications P-4
Previous Interval P-6
Private Access 2-5
Prompt Field 4-4
Properties of the Screen Painter 3-2
psplit 4-40
ptkgrp 4-40
ptrunk 4-40
pvdn 4-40
pvector 4-40

R

Range/List 4-4
Range/List Variable 4-45
Real-Time 2-6

Real-Time Database . . . P-6
Real-Time Reports . . . P-6
Real-time, Defining Report as . . . 2-6
Related Documents . . . P-8
Relational Operators . . . 4-42
Repeated Horizontally . . . 4-46
Repeated Vertically . . . 4-46
Report
  Copy . . . 3-7
  Editing with Blocks . . . 3-8
  Verify . . . 3-7
Report Access . . . 2-5
Report Description . . . 2-4
Report Input Variables . . . 4-38
Report Name . . . 2-3, 2-7
Report Select Window . . . 2-2
Report Type . . . 3-7
Reports
  Access to . . . 2-5
  Accessing the Report Select Window . . . 2-2
  Changing Input Fields for . . . 4-11
  Changing Name, Access, and Type . . . 2-8
  Changing No-Scroll Areas . . . 5-6
  Copying an Existing Report . . . 3-4
  Defining Report Type . . . 2-6
  Description, Entering . . . 2-4
  Fields in . . . 4-11
  Highlighting Fields . . . 5-1
  Input Fields for . . . 4-1
  Naming . . . 2-3
  No-Scroll Areas . . . 5-4
  Running . . . 5-20
  Saving a Design . . . 5-7
  Saving the Name . . . 2-7
  Showing Run Time/Date and User Inputs . . . 4-56
  Testing a Design . . . 5-8
  Text . . . 3-13
Reports That Cannot Be Customized . . . 1-6
Retrieve Data . . . 1-11
Reverse Video . . . 5-3
Rows
  Accessing the Row Search Window . . . 4-37
  Assigning Search ID . . . 4-51
  Changing Search Conditions . . . 4-55
  Changing Search ID . . . 4-56
  Defining . . . 4-37, 4-39, 4-42, 4-45, 4-46, 4-48, 4-49, 4-51, 6-29, 6-30, 6-31, 6-32
  Field/Bar Types for . . . 4-46
  From Tables . . . 4-39
  Range of Values . . . 6-29
  Saving Search Conditions . . . 4-51
  Selecting from a Table . . . 4-42
  Selecting Search ID . . . 4-39
  Sort Types . . . 4-48
  Sorting Data . . . 4-49
  With User Inputs . . . 4-44
Run-Date . . . 4-57
Running a Report . . . 5-20
Running Custom Reports . . . 5-20
Run-Time . . . 4-57

S

Sample of Data Ordered by Date in Ascending Order 4-49
Sample of Data Ordered by Date in Descending Order 4-48
Sample of Data Ordered by Split . . . 4-49
Sample of Data Ordered by Split and ACD Calls . 4-50
Save . . . 3-14
  Work . . . 4-60
Save the Report Name . . . 2-7
Saving
  Bar Definition . . . 4-36
  Design . . . 3-14
  Field Definitions . . . 4-27
  Report Design . . . 5-7
  Report Name . . . 2-7
  Row Search Conditions . . . 4-51
  Var/Date/Time Field . . . 4-60
  Variable Field . . . 4-9
  Your Work . . . 3-14
Saving Your Work . . . 4-60
Scale
  None . . . 4-31
  Without Tick Marks . . . 4-31
Screen Painter
  Accessing . . . 3-1
  Editing Keys . . . 3-2, 3-4
  Entering Text . . . 3-13
  Get Copy Window . . . 3-6
  Movement Keys . . . 3-2
  Moving the Cursor . . . 3-2
  Properties . . . 3-2
  Scrolling . . . 3-4
  Secondary Windows . . . 3-4
  Size . . . 3-3
  Window . . . 3-1
Scrolling Screen Painter . . . 3-4
Search ID
  Assigning . . . 4-51
  Changing . . . 4-55
  Selecting . . . 4-39
Searching for Database Items . . . 4-37
Select a Report Type . . . 3-7
Select Field . . . 4-14
Selecting Rows from an ACD . . . 4-45
Shift-Tab . . . 3-2
Shortcuts . . . 3-2
Single Value . . . 4-4
Single-User Mode . . . P-7
Size of Screen Painter . . . 3-3
Sorting Data . . . 4-49
  in Ascending or Descending Order . . . 4-48

Specific Color 5-3
spex 4-41
Split/Skill
  Status report 1-6
Status report 1-6
Step 1
  Access Row Search 4-37
  Access the Define Input Window 4-2
  Access the Get Copy Window 3-6
  Access the Report Select Window 2-2
  Access the Var/Time/Date Window 4-57
  Define the Position and Length of a Field . . . 4-13
  Deine Position and Length 4-28
Step 10
  Save the Definition 4-36
Step 2
  Assign a Name to the Report 2-3
  Define Data Expression 4-30
  Define Display Type and Format 4-57
  Define the Data Expression 4-14
  Enter a Report Name 3-6
  List the Report Input Variables 4-38
Step 3
  Define Calculation Names 4-21, 4-30
  Enter a Field Prompt 4-4
  Enter a Report Description 2-4
  Enter a User ID 3-7
  Save the Var/Date/Time Field 4-60
  Select a Row Search ID 4-39
Step 4
  Define Bar Direction 4-30
  Define Number of Field Columns and Rows . . . 4-4
  Define the User ID 2-5
  Justify Data in the Field 4-21
  Select a Report Type 3-7
  Select Tables 4-39
Step 5
  Define Access to the Report 2-5
  Define Bar Scale 4-31
  Define Field 4-4
  Select Rows in Table 4-42
  Verify the Report 3-7
Step 6
  Copy the Report 3-7
  Default Value for the Field 4-5
  Define Max Graph Value 4-33
  Define the Report as Real-Time or Historical . . 2-6
  Save the Field Definition 4-27
  Select a Field/Bar Type 4-46
Step 7
  Define First Threshold 4-34
  Save the Report Name 2-7
  Select a Sort Type 4-48
  Select Field Type 4-5
Step 8
  Associate and ACD with the Field 4-8
  Define Second Threshold 4-34
Step 9
  Normal or Reversed Thresholds 4-35
  Save a Variable Input Field 4-9
  Save Row Search 4-51
Step2
  Define Variable Name 4-2
Store Data 1-7
String-value Database Items 6-31
Subsystem P-7
Synonym Types 4-24

T

Tab 3-2
Table
  Definition 1-7
  Sample 1-8
Tables
  Building 6-4
  Creating Custom 6-1
  for Calculation Names 4-21, 4-30
  Selecting Rows from 4-39
Tasks
  1. Define the name, access, type 1-4
  2. Access Screen Painter 1-4
  3. Copy existing report 1-4
  4. Edit the report 1-4
  5. Enter report text 1-4
  6. Define the report input window 1-4
  7. Define data fields and bars 1-4
  8. Select rows of da 1-4
Terminology P-6
Test Design Error Messages 5-10
Testing a Report Design 5-8
Text Color
  Default 5-3
  Specific Color 5-3
Text, Entering 3-13
tgex 4-41
Threshold
  First 4-34
  Reversing Colors 4-35
  Second 4-34
Tick Marks 4-31
Time 4-57, 4-60
Time Formats 4-25, 4-58
Time Report Was Run 4-58
Type 2-2

U

Underline 5-3
Up 3-2
Use of Document P-2
Use of This Document P-1
User ID 2-5, 3-7
  Defining 2-5

V

V 4-28
v. 4-12
Variable 4-57, 4-60
Variable Name 4-44
  ACD 4-8
  Defining 4-2
Variable/Time/Date 4-57
vdnex 4-41
vecex 4-41
Verify a Report 3-7

W

wagent 4-40
wcwc 4-41
Weekly Data P-7
What is a Custom Report 1-2
What Is a Custom Report? 1-2
What the X's, V's, and H's Mean in a Bar 4-28
What the x's, v's, and h's Mean in a Field 4-12
Wildcard Searches 6-30, 6-31
Window
  Add Highlights 5-2
  Bar 4-29
  Define Input 4-1
  Define Input: Order 4-10
  Field 4-13
  Report Select 2-2
  Row Search 4-37
  Sample Test Design Input 5-9
  Screen Painter 3-1
  Screen Painter Get Copy 3-6
  Var/Time/Date 4-57
Windows
  Report Select 2-2
wsplit 4-40
wtkgrp. 4-41
wtrunk. 4-41
wvdn 4-41
wvector 4-41

X

X 4-28
x. 4-12

How Are We Doing?

Document Title: *CentreVu™* Call Management System Release 3 Version 5
Custom Reports

Document No.: 585-215-822 Issue 2 Date: December 1999

Lucent Technologies welcomes your feedback on this document. Your comments are of great value in helping us to improve our documentation.

1. Please rate the effectiveness of this document in the following areas:

	Excellent	Good	Fair	Poor	Not Applicable
Ease of Use					////////////////////////
Clarity					////////////////////////
Completeness					////////////////////////
Accuracy					////////////////////////
Organization					////////////////////////
Appearance					////////////////////////
Examples					////////////////////////
Illustration					
Overall Satisfaction					////////////////////////

2. Please check the ways you feel we could improve this document:

- ❒ Improve the overview/introduction
- ❒ Improve the table of contents
- ❒ Improve the organization
- ❒ Include more figures
- ❒ Add more examples
- ❒ Add more detail
- ❒ Make it more concise/brief
- ❒ Add more step-by-step procedures/tutorials
- ❒ Add more troubleshooting information
- ❒ Make it less technical
- ❒ Add more/better quick reference aids
- ❒ Improve the index

Please provide details for the suggested improvement. _____

_____

3. What did you like most about this document?

_____

_____

4. Feel free to write any comments below or on an attached sheet.

_____

_____

_____

_____

If we may contact you concerning your comments, please complete the following:

Name: _____ Telephone Number: (_____)_____

Company/Organization: _____ Date: _____

When you have completed this form, please fold, tape, and return to address on back or you can fax the form to: 303-538-2195.