CONTROL-M/Analyzer
Implementation Guide

Supporting

CONTROL-M / Analyzer version 6.2.18

July 31, 2006

Contacting BMC Software

You can access the BMC Software website at http://www.bmc.com. From this website, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

United States and Canada

Address	BMC SOFTWARE INC 2101 CITYWEST BLVD HOUSTON TX 77042-2827 USA	**Telephone**	713 918 8800 or 800 841 2031	**Fax**	713 918 8000

Outside United States and Canada

Telephone	(01) 713 918 8800	**Fax**	(01) 713 918 8000

Copyright 2006 BMC Software, Inc., as an unpublished work. All rights reserved.

BMC Software, the BMC Software logos, and all other BMC Software product or service names are registered trademarks or trademarks of BMC Software, Inc.

DB2 is a registered trademark of International Business Machines Corporation.

Oracle is a registered trademark, and the Oracle product names are registered trademarks or trademarks of Oracle Corporation.

All other trademarks belong to their respective companies.

BMC Software considers information included in this documentation to be proprietary and confidential. Your use of this information is subject to the terms and conditions of the applicable End User License Agreement for the product and the proprietary and restricted rights notices included in this documentation.

Restricted rights legend

U.S. Government Restricted Rights to Computer Software. UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure of any data and computer software by the U.S. Government is subject to restrictions, as applicable, set forth in FAR Section 52.227-14, DFARS 252.227-7013, DFARS 252.227-7014, DFARS 252.227-7015, and DFARS 252.227-7025, as amended from time to time. Contractor/Manufacturer is BMC SOFTWARE INC, 2101 CITYWEST BLVD, HOUSTON TX 77042-2827, USA. Any contract notices should be sent to this address.

Customer support

You can obtain technical support by using the Support page on the BMC Software website or by contacting Customer Support by telephone or e-mail. To expedite your inquiry, please see “Before Contacting BMC Software.”

Support website

You can obtain technical support from BMC Software 24 hours a day, 7 days a week at http://www.bmc.com/support_home. From this website, you can

- read overviews about support services and programs that BMC Software offers
- find the most current information about BMC Software products
- search a database for problems similar to yours and possible solutions
- order or download product documentation
- report a problem or ask a question
- subscribe to receive e-mail notices when new product versions are released
- find worldwide BMC Software support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support by telephone or e-mail

In the United States and Canada, if you need technical support and do not have access to the web, call 800 537 1813 or send an e-mail message to support@bmc.com. Outside the United States and Canada, contact your local support center for assistance.

Before contacting BMC Software

Before you contact BMC Software, have the following information available so that Customer Support can begin working on your problem immediately:

- product information
  - product name
  - product version (release number)
  - license number and password (trial or permanent)
- operating system and environment information
  - machine type
  - operating system type, version, and service pack or other maintenance level such as PUT or PTF
  - system hardware configuration
  - serial numbers
  - related software (database, application, and communication) including type, version, and service pack or maintenance level
- sequence of events leading to the problem
- commands and options that you used
- messages received (and the time and date that you received them)
  - product error messages
  - messages from the operating system, such as `file system full`
  - messages from related software

Contents

About This Guide 15

Conventions Used in This Guide 16
Related Publications 19

Chapter 1 Introduction to CONTROL-M/Analyzer Facilities 21

CONTROL-M / Analyzer 22
The CONTROL-M / Analyzer Tutorial 22
Preparation 23
Database Variables 24
    VIEW Option 29
    INSERT Option 30
    GRAPH Option 31
    Exiting the Database Variable Facility 32
Rules 33
    Browsing Rules 35
    Defining Rules 37
    Exiting the Rule Definition Facility 38
Rule Activity 38
    Changing Rule Activity Screen Defaults 39
    Displaying Rule Details 40
    Exploring Screen Options 41
    REPORT Option 42
    VIEW Option 43
Balancing Mission 45
    Options of the Mission List Screen 47
Active Balancing Environment 52
    Active Balancing Environment Screen 52
    SHOW Command 53
    WHY Option 55
    DISPLAY Command 56
IOA Conditions / Resources Screen 58
    Viewing Relevant Conditions 59
    Adding Conditions 59
IOA Log 63
    SHOW Command 65

Chapter 2 Example Problems and Solutions 67

Overview 68

The Example Company, INTRAC 68
INTRAC's Problems—General. 71
Sources of Data—Reports and Files 72
Basic Problems and Solutions 74
Problem 1: The Daily Invoice File — Warning 74
Problem 2: Check Payroll Totals 90
Problem 3: Check Sales Commissions. 103
Advanced Problems and Solutions 108
Problem 4: Balance Divisional Accounting Files. 108
Problem 5: Check Weekly Database Amounts 124
Problem 6: Balance Monthly Commissions 132
CONTROL-M / Analyzer Capabilities Summary. 145

Chapter 3 Preparing for Production — Application 149

Overview 150
For Newcomers to INCONTROL Products and the IOA Environment. 150
Integrated IOA Activity Log 151
Integrated Management Reporting 151
INCONTROL Products 152
Suggested Conventions. 153
Finding Data in Reports and Files 157
Saving Data as Opposed to Using Local Variables 159
Implementation Hints 161
Rule Definition 161
Quality Control and Avoiding Errors 164
Invoking CONTROL-M / Analyzer from Other Applications in the Processing Environment 166
Invoking the Rule from a Job 166
Invoking the Rule from a Balancing Mission. 166
Structured Approach for Implementation 167
Rule Worksheet A 168
Rule Worksheet B. 169
Mission Worksheet 170
Database Variable Worksheet 171

Chapter 4 Preparing for Production—Technical 173

Overview 174
CONTROL-M / Analyzer and IOA Fundamentals. 174
AutoEdit Members 175
Concurrent Usage and Locking Considerations 176
Backing Up, Restoring, Restarting 177
Use of Utilities. 178
Utility CTBJAFDL 178
Utility IOADBF. 178
Utility CTBxxxCP. 179
Utility CTBFRM 179
Setting Up a New Day Procedure 179
Purpose 179

Cleanup Activities 180
Missions 180

Appendix A Glossary **181**

Index **185**

Figures

IOA Primary Option Menu 24
CONTROL-M / Analyzer Data Base Facility Entry Panel 25
CONTROL-M / Analyzer Database Group INTRAC 26
CONTROL-M / Analyzer Database Facility List of Variables Screen 27
CONTROL-M / Analyzer Variable Definition Screen 28
Generations of Database Variable INVAMT01 29
Inserting a New Database Variable 30
Insert Database Variable Exit Prompt 31
View Graph – By Variable Generations Screen 32
CONTROL-M / Analyzer Rule Definition Entry Panel 33
List of Rules Screen 34
CONTROL-M / Analyzer Rules BROWSE Option 35
CONTROL-M / Analyzer Rule Definition Screen 36
Rule Definition Screen for a New Rule Definition 37
CONTROL-M / Analyzer Rule Activity Selection Panel 38
Rule Activity List Screen for Group INTRAC 39
Rule Activity List Screen – Display Option A 40
CONTROL-M / Analyzer Report Viewing Screen 42
Variables Committed for Rule Screen 44
CONTROL-M / Analyzer Mission Definition Entry Panel 46
Mission List Screen 46
Category List Screen 47
Missions, Divided into Categories 48
The Mission Definition Screen for Category DAILY 49
Balancing Mission Definition Exit Confirmation Window 50
CONTROL-M / Analyzer Mission Order Messages Screen 51
Active Balancing Environment Screen 52
Active Balancing Environment Show Window 53
Setting the GROUP and ENDED NOTOK Fields in the Show Window 54
Results of the Sample Show Window Selections 54
WHY List Screen 56
Active Balancing Environment Screen – Display A 57
IOA Conditions / Resources Screen 58
Viewing Relevant Conditions on the IOA / Resources Screen 59
Conditions as Prerequisites for Executing Jobs 60
ADD COND Window 61
Name and Date for a New Condition 62
Updated List of Conditions 62
IOA Log Screen 64
IOA Log Show Screen 65

CONTROL-M / Analyzer Rule Definition Entry Panel 75
New Rule Definition Screen 76
Typical Rule Structure 77
ON DATA Block Structure 78
Setting the ON Block Type 79
Flow of the ON DATA Block to Create 80
Definition of the First IF Condition 81
Defining the DO Statements in an IF Condition 82
DO SHOUT Definition 83
DO TERMINATE Definition 84
The Entire ARWARN01 Rule 85
Editing the CTBRULE1 Member 86
SYSUSER REPORT (Executed Outside Check Hours) 87
SYSPRINT — Invocation Report 88
SYSRULE Report 89
CONTROL-M / Analyzer Rule Definition Entry Panel 92
Rule Definition Screen for Rule ARWARN02 93
INPUT Block 94
ON Block Structure 95
The EXECUTE CHECK ON Block 98
IF and ELSE Parameters 99
ON Block ENDJOB 100
Output from Job RUNRULE2 102
Rule Definition Screen for Rule ARWARN03 104
ON Block INPUT 105
ON Block EXECUTE FINISH 106
SYSUSER Report from RUNRULE3 107
Rule ARWARN04 109
EXECUTE INIT Block 110
EXECUTE INPUT Block 111
Sample Data for DO EXTRACT Definition 112
DO SET Parameters 114
ON Block EXECUTE CHKDETL 116
End of Rule ARWARN04 117
Flow of ON Block Logic 118
ON Block EXECUTE ENDJOB 119
Analysis on Cost Amounts 120
LABEL: FINISH 121
SYSUSER Report from RUNRULE4 123
SYSUSER Report from RUNRUL4A 124
Rule ARWARN05 127
LABEL ACCUM in the ACCUMSLS ON Block 128
Balancing Checks of the ON Block EXECUTE ACCUMSLS 129
Last Balancing Checks of the ON Block EXECUTE ACCUMSL 130
End of Rule ARWARN05 131
SYSUSER Report from RUNRULE5 131
Rule ARWARN06 133
EXECUTE INPUT ON Block 134
Second Compound Grouping of WHEN Parameters 136

ON Block CHKNAME 137
ON Block FInish 139
Sample Output Produced by the RUNRULE6 Job 140
Rule ARWARN06 EXECUTE FINISH Block (Display 1) 141
Rule ARWARN06 EXECUTE FINISH Block (Display 2) 142
Rule ARWARN06 EXECUTE FINISH Block (Display 3) 143
Rule ARWARN06 LABEL: ENDJOB 144
Sample Output Produced by LABEL: ENDJOB 145
ON Block Structure Summary 146
Data Sources That ON Statements Can Process 147
What Automated Balancing Action Parameters
(DO Statements) Can Do 148

Tables

Procedures and Libraries for Your Installation 23
Parameters that Define a Database Variable 28
Fields of the IOA Log Screen 64
File Format for Daily Invoice File 72
Fields to Define a New Rule 76
List of INCONTROL Products 152
Glossary of Terms 181

About This Guide

This guide is comprised of the following chapters:

Chapter 1 – Introduction to CONTROL-M/Analyzer Facilities

An online demonstration of basic CONTROL-M / Analyzer functionality

Chapter 2 – Example Problems and Solutions

A case study of a sample organization, its control and balancing needs, and the CONTROL-M / Analyzer solutions

Chapter 3 – Preparing for Production - Application

Suggestions and recommendations for the implementation of CONTROL-M / Analyzer applications

During the implementation process you typically work with operations personnel and application analysts.

Chapter 4 – Preparing for Production - Technical

Suggestions and recommendations to the installer and maintainer of the CONTROL-M / Analyzer environment

This typically involves the systems programmer, working in combination with operations personnel.

Conventions Used in This Guide

Notational conventions that may be used in this guide are explained below.

Standard Keyboard Keys

Keys that appear on the standard keyboard are identified in boldface, for example, **Enter, Shift, Ctrl+S** (a key combination), or **Ctrl S** (a key sequence).

WARNING

The commands, instructions, procedures, and syntax illustrated in this guide presume that the keyboards at your site are mapped in accordance with the EBCDIC character set. Certain special characters are referred to in this documentation, and you must ensure that your keyboard enables you to generate accurate EBCDIC hex codes. This is particularly true on keyboards that have been adapted to show local or national symbols. You should verify that

$ is mapped to x'5B'
# is mapped to x'7B'
@ is mapped to x'7C'

If you have any questions about whether your keyboard is properly mapped, contact your system administrator.

Preconfigured PFKeys

Many commands are preconfigured to specific keys or key combinations. This is particularly true with regard to numbered PF keys, or pairs of numbered PFKeys. For example, the END command is preconfigured to, and indicated as, **PF03/PF15**. To execute the END command, press either the **PF03** key or the **PF15** key.

Instructions to enter commands may include

- only the name of the command, such as, enter the END command
- only the PF keys, such as, press **PF03/PF15**
- or both, such as, press **PF03/PF15,** or enter the END command

Command Lines and Option Fields

Most screens contain a command line, which is primarily used to identify a single field where commands, or options, or both, are to be entered. These fields are usually designated COMMAND, but they are occasionally identified as COMMAND / OPT or COMMAND / OPTION.

Option field headings appear in many screens. These headings sometimes appear in the screen examples as OPTION, or OPT, or O.

Names of Commands, Fields, Files, Functions, Jobs, Libraries, Members, Missions, Options, Parameters, Reports, Subparameters, and Users

The names of commands, fields, functions, jobs, libraries, members, missions, options, parameters, reports, subparameters, users, and most files, are shown in standard UPPERCASE font.

User Entries

In situations where you are instructed to enter characters using the keyboard, the specific characters to be entered are shown in this **UPPERCASE BOLD** text, for example, type **EXITNAME.**

Syntax statements

In syntax, the following additional conventions apply:

- A vertical bar (|) separating items indicates that you must choose one item. In the following example, you would choose *a*, *b*, or *c*:

  `a| b | c`

- An ellipsis (. . .) indicates that you can repeat the preceding item or items as many times as necessary.

- Square brackets ([ ]) around an item indicate that the item is optional. If square brackets ([ ]) are around a group of items, this indicates that the item is optional, and you may choose to implement any single item in the group. Square brackets can open ([ ) and close ( ]) on the same line of text, or may begin on one line of text and end, with the choices being stacked, one or more lines later.

- Braces ({ }) around a group of items indicates that the item is mandatory, and you must choose to implement a single item in the group. Braces can open ({) and close (}) on the same line of text, or may begin on one line of text and end, with the choices being stacked, one or more lines later.

Screen Characters

`All syntax, operating system terms, and literal examples are presented in this typeface.` This includes JCL calls, code examples, control statements, and system messages. Examples of this are:

- calls, such as

```
CALL 'CBLTDLI'
```

- code examples, such as

```
FOR TABLE owner.name USE option, . . . ;
```

- control statements, such as

```
//PRDSYSIN DD * USERLOAD PRD(2) PRINT
```

- system messages, both stand-alone, such as `You are not logged on to database database_name`, and those embedded in text, such as the message `You are not logged on to database database_name`, are displayed on the screen.

Variables

Variables are identified with *italic* text. Examples of this are:

- In syntax or message text, such as
  `Specify database` *database_name*
- In regular text, such as
  replace database *database_name1* with database *database_name2* for the current session
- In a version number, such as
  EXTENDED BUFFER MANAGER for IMS 4.1.*xx*

Special elements

This book includes special elements called *notes* and *warnings*:

NOTE

Notes provide additional information about the current subject.

WARNING

Warnings alert you to situations that can cause problems, such as loss of data, if you do not follow instructions carefully.

Related Publications

Additional CONTROL-M / Analyzer functions and facilities that are not covered in this guide are described in the following manuals:

CONTROL-M/Analyzer User Guide

Detailed explanation of the base CONTROL-M / Analyzer product

INCONTROL for z/OS Installation Guide

Step-by-step guide to installing INCONTROL products using the INCONTROL™ Installation and Customization Engine (ICE) application

INCONTROL for z/OS Administrator Guide

Information for system administrators about customizing and maintaining INCONTROL products

INCONTROL for z/OS Security Guide

Step-by-step guide to implementing security in INCONTROL products

Chapter 1

Introduction to CONTROL-M/Analyzer Facilities

This chapter includes the following topics:

CONTROL-M / Analyzer 22
The CONTROL-M / Analyzer Tutorial 22
Preparation 23
Database Variables 24
VIEW Option 29
INSERT Option 30
GRAPH Option 31
Exiting the Database Variable Facility 32
Rules 33
Browsing Rules 35
Defining Rules 37
Exiting the Rule Definition Facility 38
Rule Activity 38
Changing Rule Activity Screen Defaults 39
Displaying Rule Details 40
Exploring Screen Options 41
REPORT Option 42
VIEW Option 43
Balancing Mission 45
Options of the Mission List Screen 47
Active Balancing Environment 52
Active Balancing Environment Screen 52
SHOW Command 53
WHY Option 55
DISPLAY Command 56
IOA Conditions / Resources Screen 58
Viewing Relevant Conditions 59
Adding Conditions 59
IOA Log 63
SHOW Command 65

CONTROL-M/Analyzer

CONTROL-M / Analyzer is a flexible, automated tool providing complete quality control in a modern information processing environment. Within a completely integrated facility, the powerful CONTROL-M / Analyzer functions analyze and balance data, and immediately report errors and inconsistencies.

CONTROL-M / Analyzer is the professional tool for operations staff, end users, analysts, and managers. Coupled with other INCONTROL® products, CONTROL-M / Analyzer makes automated operations a reality. End user concerns about data integrity are resolved with the extensive data manipulation and checking capabilities of CONTROL-M / Analyzer.

The CONTROL-M/Analyzer Tutorial

In the framework of a hands-on demonstration, you will be guided through the major online functions of CONTROL-M / Analyzer. In addition, you will have an opportunity to go through a structured learning exercise to become familiar with CONTROL-M / Analyzer rules. By learning and practicing with these rules, in combination with the online functions of CONTROL-M / Analyzer, you will solve the control and balancing problems of a sample organization.

The tutorial in this chapter introduces you to the following CONTROL-M / Analyzer facilities:

- Database Variables
- Rules
- Rule Activity
- Balancing Missions
- Active Balancing Environment
- IOA Conditions / Resources Screen
- IOA Log

Preparation

Begin by performing the following procedures to ensure that the Getting Started environment is installed correctly and ready for your use.

1. To set up the Getting Started environment, ensure that you have run step 8.6, Getting Started, using the INCONTROL Installation and Customization Engine (ICE) application. This step displays member PREPGSB for editing, and runs the resultant job.

   After the job has executed successfully, you are ready to proceed. If you want to investigate the results of this job, contact your system administrator or refer to the *INCONTROL for z/OS Installation Guide* for instructions.

2. If your system administrator has provided you with procedures and library names that are different from the CONTROL-M / Analyzer Installation default, note them here for future reference:

Table 1 Procedures and Libraries for Your Installation

Procedure or Library	Installation Non-Default Name
TSO Logon Procedure	
RULE Library	
MISSION Library	
JCL Library	
Sample Reports Library	

3. For any members that reside in the IOA SAMPLE library, you must do the following in order for them to be resolved correctly:

   A. Copy the member to a work library.

   B. Edit and submit the IOAINSJ job from the IOA JCL library. For more information on this job, see the *INCONTROL for z/OS Administrator Guide.*

4. Enter the IOA Primary Option Menu, using any online environment supported by CONTROL-M / Analyzer, such as TSO, TSO / ISPF, ROSCOE, CICS, IMS / DC, VTAM, IDMS, or COM-PLETE, or use the procedure provided by your system administrator.

Figure 1 IOA Primary Option Menu

```
---------------------       IOA PRIMARY OPTION MENU       ------------------(1)
 OPTION ===>                                                 USER        N06

 IOA                        CONTROL-D/V                CONTROL-O

   4  COND/RES                A  MISSION STATUS         OR  RULE DEFINITION
   5  LOG                     M  MISSION DEF            OM  MSG STATISTICS
   6  TSO                     R  REPORT DEF             OS  RULE STATUS
   7  MANUAL COND             T  RECIPIENT TREE         OL  AUTOMATION LOG
   8  CALENDAR DEF            U  USER REPORTS           OA  AUTOMATION OPTS
  IV  VARIABLE DATABASE       F  PC PACKET STATUS       OC  COSMOS STATUS
                             DO  OBJECTS                OK  KOA RECORDER


 CONTROL-M & CTM/Restart    CONTROL-M/Analyzer         CONTROL-M/Tape

   2  JOB SCHEDULE DEF       BB  BALANCING STATUS       TR  RULE DEFINITION
   3  ACTIVE ENV.            BM  MISSION DEF            TP  POOL DEFINITION
   C  CMEM DEFINITION        BV  DB VARIABLE DEF        TV  VAULT DEFINITION
                             BR  RULE DEFINITION        TI  INQ/UPD MEDIA DB
                             BA  RULE ACTIVITY          TC  CHECK IN EXT VOL


 COMMANDS:  X - EXIT, HELP, INFO  OR CHOOSE A MENU OPTION              15.48.22
```

The menu that you see may have other options in addition to those shown above. This depends on how your computer center has set up the CONTROL-M / Analyzer and INCONTROL environment.

Database Variables

The first facility discussed is the CONTROL-M / Analyzer Database facility.

1 Select option BV and press **Enter**. The **CONTROL-M/Analyzer** Database Facility entry panel is displayed.

Figure 2 CONTROL-M/Analyzer Data Base Facility Entry Panel

```
---------- CONTROL-M/Analyzer DATA BASE FACILITY - ENTRY PANEL --------------(BV)
COMMAND ===>



SPECIFY GROUP, VARIABLE NAME


GROUP    ===>                            (Blank for Group selection list)

VARIABLE ===>                            (Blank for Variable selection list)

                                         ("*" for Variable/Group prefix)









USE THE COMMAND "SHPF" TO SEE PFK ASSIGNMENT                          08.09.34
```

In CONTROL-M / Analyzer, work is organized into groups. A group usually contains CONTROL-M / Analyzer balancing jobs and variables with a common purpose. A group is functional in nature, and helps reflect actual divisions in the workplace. Some group examples are: ACCOUNTING, INVENTORY, MIS_DEPT.

Use of groups within CONTROL-M / Analyzer facilitates the work and provides several important features:

- **Looking at Rule Activity and the Active Balancing File** – Rules can be listed according to a specific group. Only relevant data are shown, providing quick access to needed information.

- **Reviewing the IOA Log** – Same as “Looking at Rule Activity and the Active Balancing File.”

- **Group Feature Common with CONTROL-M Jobs and CONTROL-D Missions** – This common feature between CONTROL-M / Analyzer rules, CONTROL-M jobs, and CONTROL-D missions makes it easy to track an application throughout the IOA environment.

This guide now examines a subset of the CONTROL-M / Analyzer database—a group defined for the example company, INTRAC.

NOTE

To provide a comprehensive and realistic illustration of CONTROL-M / Analyzer rules, the examples in this guide use a fictional company called the Intelligent Transport Company (INTRAC). Any resemblance to actual organizations in existence is purely coincidental and unintentional.

2 Place the cursor in the GROUP field.

3 Type **INTRAC**.

Figure 3 CONTROL-M/Analyzer Database Group INTRAC

```
------------ CONTROL-M/Analyzer DATA BASE FACILITY - ENTRY PANEL ------------(BV)
COMMAND ===>



SPECIFY GROUP, VARIABLE NAME


GROUP    ===> INTRAC                    (Blank for Group selection list)

VARIABLE ===>                           (Blank for Variable selection list)

                                        ("*" for Variable/Group prefix)









USE THE COMMAND "SHPF" TO SEE PFK ASSIGNMENT                          08.09.34
```

4 Ensure that the VARIABLE field is blank and press **Enter**.

When the variable field is left blank, CONTROL-M / Analyzer shows a list of all database variables defined for the group INTRAC.

Figure 4 CONTROL-M/Analyzer Database Facility List of Variables Screen

```
LIST OF VARIABLES   CONTROL-M/Analyzer DATA BASE    GROUP: INTRAC
COMMAND ===>                                                SCROLL===>CRSR
OPT NAME ---------------------------------------------------------------------
    COST01
    COST02
    COST03
    INVAMT01
    INVAMT02
    INVAMT03
    PROCESS_MONTH
===== >>>>>>>>>>>>>>>>>>   NO MORE VARIABLES IN GROUP  <<<<<<<<<<<<<<<<< =====



OPTIONS:  S SELECT   I INSERT   V VIEW     G GRAPH                    08.09.34
```

Notice the OPT field on the left side of the screen. Available options that can be typed in this OPT field are listed on the bottom of the screen. Each option performs an action on the associated database variable.

5 Place the cursor in the OPT field, to the left of the field with variable name INVAMT01.

6 Type **S** and press **Enter** to select this variable. The **CONTROL-M/Analyzer** Variable Definition screen for the variable INVAMT01 is displayed.

Figure 5 CONTROL-M/Analyzer Variable Definition Screen

```
------------------  CONTROL-M/Analyzer VARIABLE DEFINITION  -------------(BV.S)
COMMAND ===>                                                   SCROLL===> CRSR
+-------------------------------------------------------------------------------+
| NAME     INVAMT01               GROUP INTRAC                                  |
| CREATED  08/08/00 - 09:22:04    BY    N61A      TYPE    (B-BATCH,O-ONLINE)    |
| DESC     DAILY-INVOICES-DIVISION-1                                            |
| ============================================================================ |
| MAXIMUM ACTIVE GENERATIONS 0025   GEN TYPE I (I-INDEPENDENT,H-HIERARCHICAL)   |
| CURRENT ACTIVE GENERATIONS 0006                                              |
| OLDEST   -5    DATE 08/08/00                                                 |
| LATEST    0    DATE 08/08/00                                                 |
| ============================================================================ |
| TYPE     VAR                                                                 |
| MASK                                                                         |
|                                                                              |
============= >>>>>>>>> BOTTOM OF VARIABLE DEFINITION <<<<<<<<<<< =============



 PLEASE FILL IN VARIABLE DEFINITION                                   08.14.59
```

Table 2 Parameters that Define a Database Variable

Parameter	Description	Value in Example
NAME	A name for the variable.	**INVAMT01**
GROUP	The group to which it belongs	**INTRAC**
DESC	A description of the variable	**DAILY-INVOICES-DIVISION-1**
MAXIMUM ACTIVE GENERATIONS	The number of generations of data to maintain.	**25**

NOTE

The Maximum Active Generations keeps track of up to 1000 generations or versions of a variable. You may choose to keep multiple generations to be able to go back in time and recover data, or to reference multiple versions of data. For an example this feature see “Problem 5: Check Weekly Database Amounts” on page 2-124.

CONTROL-M / Analyzer automatically maintains the other information in the parameters on this screen. For a description of the other parameters on this screen, or other screens throughout this guide, refer to the *CONTROL-M/Analyzer User Guide*.

VIEW Option

In this exercise, you will learn about the List of Variables screen VIEW option.

1. Press **PF03/PF15**, corresponding to the END command, to return to the List of Variables screen.

2. Place the cursor in the OPT field to the left of the same variable INVAMT01, and type **V**.

3. Press **Enter** to view this variable. The following screen appears with the associated generations, or versions, of the Database Variable INVAMT01:

Figure 6 Generations of Database Variable INVAMT01

```
-------<G>-------- GROUP  INTRAC        VAR  INVAMT01            -------(BV.V)
COMMAND ===>                                                   SCROLL===> CRSR
O GROUP NAME --------- VARIABLE NAME ------ GENERATION ------------------------
  INTRAC               INVAMT01             000
   Value         : 367014
   User Key      :                        Run Id  :
   Creation Date : 08/08/00   12:03:48   By User : M66      Type : B
-------------------------------------------------------------------------------
  INTRAC               INVAMT01             001
   Value         : 367014
   User Key      :                        Run Id  :
   Creation Date : 08/08/00   10:58:37   By User : M66      Type : B
-------------------------------------------------------------------------------
  INTRAC               INVAMT01             002
   Value         : 367014
   User Key      :                        Run Id  :
   Creation Date : 08/08/00   10:55:59   By User : M66      Type : B
-------------------------------------------------------------------------------
  INTRAC               INVAMT01             003
   Value         : 367014
   User Key      :                        Run Id  :
   Creation Date : 08/08/00   10:34:10   By User : M66      Type : B
-------------------------------------------------------------------------------
OPTIONS:  U UPDATE D DELETE                                            08.16.35
```

In this example, INVAMT01 has the following four generations: 000, 001, 002, and 003. Each generation of INVAMT01 includes the following information: a value, displayed in the Value field, creation date and time each generation of INVAMT01 was created, displayed in the Creation Date field, and a user name for the user that created the variable, displayed in the By User field.

NOTE

Users with security clearance can update, delete or add information on this screen.

These multiple generations of the variable ensure that a CONTROL-M / Analyzer rule can be rerun. CONTROL-M / Analyzer can restore the data as it existed at a prior date and time (that is, an earlier generation of the variable) before a database update has occurred, providing an audit trail for the Database Variable.

You can create a new generation or edit an existing generation as necessary, as illustrated in Chapter 2.

INSERT Option

In this exercise, you will learn about the List of Variables screen INSERT option to insert a new variable into the CONTROL-M / Analyzer database.

1. Press **PF03/PF15** to return to the List of Variables screen.

2. Place the cursor in the OPT field to the left of variable INVAMT01, and type **I**.

3. Press **Enter** to insert a new variable. The following screen is displayed with the variable name in the NAME field blanked out. Notice that other information about the variable, such as the group name, displayed in the GROUP field, description, displayed in the DESC field, and maximum generations, displayed in the MAXIMUM ACTIVE GENERATIONS field, is already filled in.

Figure 7 Inserting a New Database Variable

```
------------------  CONTROL-M/Analyzer VARIABLE DEFINITION  ---------------(BV.S)
COMMAND ===>                                                   SCROLL===> CRSR
+-------------------------------------------------------------------------+
| NAME                            GROUP INTRAC                            |
| CREATED  11/12/00 - 08:18:12    BY    M86       TYPE  O (B-BATCH,O-ONLINE)  |
| DESC     DAILY-INVOICES-DIVISION-1                                      |
| ======================================================================= |
| MAXIMUM ACTIVE GENERATIONS 0025   GEN TYPE I (I-INDEPENDENT,H-HIERARCHICAL) |
| CURRENT ACTIVE GENERATIONS                                              |
| OLDEST         DATE                                                     |
| LATEST         DATE                                                     |
| ======================================================================= |
| TYPE     VAR                                                            |
| MASK                                                                    |
|                                                                         |
============= >>>>>>>>> BOTTOM OF VARIABLE DEFINITION <<<<<<<<<<< =============



 PLEASE FILL IN VARIABLE DEFINITION                                  08.18.12
```

When you insert a new variable, CONTROL-M/Analyzer copies the definition information from the Database Variable where your cursor had been positioned on the List of Variables screen. In this case, the information has been copied from variable INVAMT01. This feature allows you to save many keystrokes when you are creating a new variable from another variable that has a similar definition. Once created, you can modify the information in the new Database Variable as needed.

4 Enter the variable name **TOTAL1** in the NAME field.

5 Press **PF03/PF15** to exit this function. The following prompt is displayed:

Figure 8 Insert Database Variable Exit Prompt

```
-------------------  CONTROL-M/Analyzer VARIABLE DEFINITION  --------------(BV.S)
COMMAN +-------------------------------------------------------+  ===> CRSR
+----- |                 PLEASE SELECT EXIT OPTION                |  --------+
| NAME |                                                          |          |
| CREA |     SAVE     CREATE                                      |  NLINE)  |
| DESC |                                                          |          |
| ==== |     VARIABLE TOTAL1                                      |  ======= |
| MAXI |                                                          |  CHICAL) |
| CURR +-------------------------------------------------------+          |
| OLDEST         DATE                                                      |
| LATEST         DATE                                                      |
| ========================================================================= |
| TYPE     VAR                                                             |
| MASK                                                                     |
|                                                                          |
============= >>>>>>>>> BOTTOM OF VARIABLE DEFINITION <<<<<<<<<<< =============





 PLEASE FILL IN VARIABLE DEFINITION                                  08.18.12
```

6 By entering **Y** in the CREATE field, you would actually create a new variable called TOTAL1 in the database. Because you do not require this variable, type **N** in the CREATE field and press **Enter**.

GRAPH Option

In this exercise, you will learn about the List of Variables screen GRAPH option.

1 Place the cursor in the OPT field to the left of the INVAMT01 variable, and type **G**.

2 Press **Enter**. The View Graph – By Variable Generations screen is displayed.

Figure 9 View Graph – By Variable Generations Screen

```
-------------------- VIEW GRAPH - BY VARIABLE GENERATIONS  --------------(BV.G)
COMMAND ===>                                                  SCROLL===> CRSR
SCALE  1 : 7341      GROUP  INTRAC              VARIABLE  INVAMT01
  GENERATION        VALUE   +---+---10----+---20----+---30----+---40----+---50
   GEN 000         367014
   GEN 001         367014
   GEN 002         367014
   GEN 003         367014
   GEN 004         367014
   GEN 005         367014
====== >>>>>>>>>>>>>>>>   NO MORE ENTRIES IN THE LIST     <<<<<<<<<<<<<<<< ======

COMMANDS:   REFRESH (VIEW DATA)   END (RETURN TO PREVIOUS SCREEN)      08.28.58
```

In this example, each of the INVAMT01 generations is shown. Each line includes the GENERATION number, its VALUE, and space for a graphical representation of changes to the value from generation to generation. In this case, because each generation of INVAMT01 has the same value, no graphical representation is shown.

The scale of the graph can be changed and the entire screen can be refreshed. For more information on this feature, see the discussion of the View Graph by Variable Generations screen in the *CONTROL-M/Analyzer User Guide*.

Exiting the Database Variable Facility

You can return to the Primary Option menu in the following ways:

- Pressing **PF03/PF15** (or entering the END command) and exiting each screen until you reach the Primary Option menu.

- Using the Multi-Screen Control feature. This allows CONTROL-M/Analyzer you to go from one function directly to another without having to go through the IOA Primary Option Menu.

  When going directly from one function to another, the exact location and data on the screen you are exiting from are stored. If you return to the function using the Multi-Screen Control feature, you are returned to the same screen in the same condition as when you left it.

You use the Multi-Screen Control feature by placing the cursor in the COMMAND line, typing an equal sign (=), typing the IOA Primary Option Menu option, and then pressing **Enter**. For example, entering **=BV** brings you to the Database Variable Definition screen.

This feature of CONTROL-M / Analyzer can be used to jump immediately to any of the online menu options. You will now use this feature to jump directly to the Primary Option menu.

1 In the COMMAND line, at the top of the screen, type **=1**.

2 Press **Enter** to return directly to the IOA Primary Option menu.

You have now completed an online review of the Database facilities of CONTROL-M / Analyzer. The tutorial continues with a brief look at CONTROL-M / Analyzer rules.

Rules

In this exercise you will locate rules, learn what information is available about rules, and learn what a rule looks like. In Chapter 2, you will examine rules in more depth to learn how to build them in CONTROL-M / Analyzer.

1 From the Primary Option menu, type **BR** and press **Enter**. The **CONTROL-M/Analyzer** Rule Definition entry panel is displayed.

Figure 10 CONTROL-M/Analyzer Rule Definition Entry Panel

```
------------------- CONTROL-M/Analyzer RULE DEFINITION - ENTRY PANEL ------(BR)
COMMAND ===>



SPECIFY LIBRARY, RULE

   LIBRARY ===> CTB.PROD.RULES
   RULE    ===>                      (Blank for rule selection list)



USE THE COMMAND "SHPF" TO SEE PFK ASSIGNMENT                          08.40.15
```

In the Rule Definition entry panel you will examine the library of rules which exists for the example company, INTRAC.

2 If the library name in the LIBRARY field in this example is different from your CONTROL-M / Analyzer RULES library name, correct the example. The RULE field should be blank.

3 Press **Enter** to display the List of Rules screen, which displays the rules in the CONTROL-M / Analyzer RULES library.

Figure 11 List of Rules Screen

```
LIST OF RULES  IN CTB.PROD.RULES                                ------------(BR)
COMMAND ===>                                                     SCROLL===> CRSR
OPT  NAME ------------ VV.MM  CREATED     CHANGED     SIZE  INIT   MOD   ID
     ARSETVAR          01.14 00/09/10 00/11/30 16:55    31    18     0 N14
     ARWARN01          01.00 00/11/12 00/11/03 13:57    34    34     0 N14
     ARWARN02          01.56 00/08/05 00/11/07 14:43    54    46     0 N19A
     ARWARN03          01.48 00/08/05 00/11/08 10:50    34    46     0 M32A
     ARWARN04          01.66 00/09/03 00/11/27 17:48   122    17     0 N34
     ARWARN05          01.11 00/09/10 00/09/27 13:51    78    49     0 N34
     ARWARN06          01.69 00/08/05 00/11/12 12:12    97    46     0 N19
     ARWARN4A          01.66 00/09/05 00/11/12 12:12   122    17     0 N19

OPTIONS:  S SELECT   B BROWSE   D DELETE  I INSERT  N NEW              08.41.23
```

Locate the rules shown in the above example in the list of rules on your screen.

NOTE

If your list spans more than one screen, you can press the standard **PF07/PF19** (UP) and **PF08/PF20** (DOWN) keys to scroll forward and backward through the list of rules.

By default, only member names are listed on the screen. However, if the default has been modified at time of installation, statistical information is displayed for each member name, as shown in the screen example above.

The screen has several options for performing functions on any of the rules displayed. These options are shown at the bottom of the screen, including:

- S (SELECT) – Select a rule to update
- B (BROWSE) – Select a rule to browse
- D (DELETE) – Delete a rule
- I (INSERT) – Insert a new rule, which is a copy of an existing rule
- N (NEW) – Create a new rule

Browsing Rules

1 Place the cursor in the OPT field to the left of the rule named ARWARN01, and type **B**.

Figure 12 CONTROL-M/Analyzer Rules BROWSE Option

```
LIST OF RULES  IN CTB.PROD.RULES                                ------------(BR)
COMMAND ===>                                                     SCROLL===> CRSR
OPT  NAME ------------ VV.MM  CREATED     CHANGED     SIZE  INIT   MOD   ID
     ARSETVAR          01.14 00/09/10 00/11/30 16:55    31    18     0 N14
 B   ARWARN01          01.00 00/11/12 00/11/03 13:57    34    34     0 N14
     ARWARN02          01.56 00/08/05 00/11/07 14:43    54    46     0 N19A
     ARWARN03          01.48 00/08/05 00/11/08 10:50    34    46     0 M32A
     ARWARN04          01.66 00/09/03 00/11/27 17:48   122    17     0 N34
     ARWARN05          01.11 00/09/10 00/09/27 13:51    78    49     0 N34
     ARWARN06          01.69 00/08/05 00/11/12 12:12    97    46     0 N19
     ARWARN4A          01.66 00/09/05 00/11/12 12:12   122    17     0 N19









OPTIONS:  S SELECT   B BROWSE   D DELETE  I INSERT  N NEW              08.41.23
```

2 Press **Enter** to browse this rule. The details of the rule are displayed on the CONTROL-M / Analyzer Rule Definition screen.

Figure 13 CONTROL-M/Analyzer Rule Definition Screen

```
        LIBRARY : CTBP.PROD.RULES                                RULE : ARWARN01
COMMAND ===>                                                     SCROLL===> CRSR
+-------------------------------------------------------------------------------+
| OWNER   M66                      GROUP INTRAC                                 |
| UPDATED 24/04/00  -  12:00:39    BY IOA6100                                   |
| DESC    SHOUT TO USERS IF DIVISIONAL AR DATA NOT IN                           |
| OPTIONS                                                                       |
| ============================================================================= |
| EXECUTE CHECK    UPON                                                     C    |
| ON DATA                                                                       |
| IF       %%SYSTIME >= 120000 AND %%SYSTIME < 130000                      C    |
|   DO PRINT    = DIV. INVOICES NOT IN, PLEASE CHECK IMMEDIATELY      F   C    |
|   DO PRINT    = ********* ACTION REQUIRED WITHIN 30 MINUTES ******* F   C Y  |
|             **                                                                |
|   DO SHOUT    TO TSO-M66             URGENCY U                              |
|    = DIV. INVOICES NOT IN, PLEASE CHECK IMMEDIATELY *************            |
|   DO TERMINAT = NOTOK    COD 0009                                             |
| IF       %%SYSTIME >= 130000 AND %%SYSTIME < 150000                      C    |
|   DO PRINT    = DIV. INVOICES STILL NOT IN, URGENT!!!               F   C    |
|   DO PRINT    = ***************** HELP NEEDED NOW ***************** F   C Y  |
|             **                                                                |
|   DO SHOUT    TO TSO-M66             URGENCY U                              |
|    = DIV. INVOICES STILL NOT IN, URGENT ACTION REQUIRED*********            |
PLEASE FILL IN RULE DEFINITION.                                          14.18.42
```

CONTROL-M / Analyzer uses rule definitions to examine files and reports in detail, to analyze data, and to take various actions based on the balancing execution results.

Rule Definition Screen Descriptive Information

Descriptive information for the rule appears on the top half of the screen, including

- the LIBRARY in which the rule is located

  In this case, the library name is CTB.PROD.RULES

- the RULE name

  In this case, the rule name is ARWARN01.

- the OWNER of the rule

  In this case, the rule owner is M66.

- the GROUP to which the rule belongs

  In this case, the group name is INTRAC.

- Other descriptive information appears in the DESC and OPTION fields.

Rule Definition Screen Instructions

The rule instructions begin with an EXECUTE statement. Here you program CONTROL-M / Analyzer to input data from files or reports, extract information, and execute balancing actions.

The rule example in Figure 13 is programmed to

- examine the system time
- print a message based on the result, and send a Shout message to an online user

Defining Rules

In Chapter 2, you will examine several rules and then define a new rule yourself. This exercise briefly illustrates how to define a rule.

1. Press **PF03/PF15** to return to the List of Rules screen.
2. Place the cursor in the OPT field to the left of the rule named ARWARN01.
3. Type **N** and press **Enter**. The Rule Definition screen is displayed.

Figure 14 Rule Definition Screen for a New Rule Definition

```
CTB303E FILL IN THE REQUIRED FIELD                              RULE :
COMMAND ===>                                                    SCROLL===> CRSR
+-------------------------------------------------------------------------------+
| OWNER                              GROUP                                      |
| UPDATED 11/12/00  -  08:56:08    BY M86                                       |
| DESC                                                                          |
| OPTIONS                                                                       |
| ============================================================================ |
| EXECUTE          UPON                                                      C   |
| ON                                                                            |
======= >>>>>>>>>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<<<<<< =====

PLEASE FILL IN RULE DEFINITION.                                          08.56.08
```

Exiting the Rule Definition Facility

To exit the Rule Definition facility at this time, you must cancel the creation of the new rule you just created as follows:

1. Place the cursor in the COMMAND field.
2. Type **CANCEL**, and press **Enter**. You are returned to the List of Rules screen.

Rule Activity

You can use the CONTROL-M/Analyzer Rule Activity facility to obtain information about a job which executes a rule.

1. If you are not working in the Primary Option menu, enter **=1** at the COMMAND field.
2. From the Primary Option Menu, enter **BA** in the OPTION field. The **CONTROL-M/Analyzer** Rule Activity selection panel is displayed.

Figure 15 CONTROL-M/Analyzer Rule Activity Selection Panel

```
--------------------- CONTROL-M/Analyzer - RULE ACTIVITY --------------------(BA)
COMMAND/OPTION  ===>

SELECT BY:
   GROUP        ===> INTRAC                Group name or * for all groups
   USERID       ===>                       User name or prefix
   JOBNAME      ===>                       Job name or prefix
   STEPNAME(S)  ===>                       Step name(s)
   REMARK       ===>                       Remark or prefix
   RULE         ===>                       Rule name
   CATEGORY     ===>                       Site-defined category or mask
   DATE FROM    ===>        TO             Job date
   TIME FROM    ===>        TO             Job time

SELECT BY STATUS:
   OK           ===> Y
   Not OK       ===> Y
   In tolerance ===> Y

DISPLAY OPTIONS:
   Bypass panel ===> N
   Display type ===> A

FILL IN THE SELECTION CRITERIA, AND PRESS ENTER                        08.57.48
```

NOTE

It is preferable to enter the Rule Activity facility through the selection panel, rather than going directly to the Rule Activity screen using the Multi-Screen Control feature. Because the Multi-Screen Control feature often displays a large amount of information about all of the active CONTROL-M / Analyzer rules, you would normally want to use the Rule Activity selection panel to display only a specific rule or group of rules.

Changing Rule Activity Screen Defaults

In this exercise, you will change the Rule Activity screen default display options so that you see only rules associated with the group INTRAC.

1. Type **INTRAC** in the GROUP field.

2. Type **D** (for default display) in the DISPLAY TYPE field and press **Enter**. The Rule Activity List screen is displayed only for those rules in the INTRAC group.

Figure 16 Rule Activity List Screen for Group INTRAC

```
RULE ACTIVITY  <D>  GROUP INTRAC               USERID          JOBNAME
COMMAND ===>                                                    SCROLL===> CRSR
O JOBNAME  JOBID STEPNAME          DATE     START ENDED RULE     STATUS  CODE
  I610INGS 24883 STEP4    CONTROLB 07/12/00 12:03 12:03 ARSETVAR OK      0000
  I610INGS 24883 STEP5    CONTROLB 07/12/00 12:03 12:03 ARWARN01 OK      0000
  I610INGS 24883 STEP6    CONTROLB 07/12/00 12:04 12:04 ARWARN04 NOTOK   0999
  I610INGS 24883 STEP7    CONTROLB 07/12/00 12:04 12:04 ARWARN4A OK      0009
  I610INGS 24883 STEP8    CONTROLB 07/12/00 12:04 12:04 ARWARN01 OK      0000
====== >>>>>>>>>>>>>>>>>   B O T T O M    O F    L I S T   <<<<<<<<<<<<<< ======



OPTIONS:   V VIEW L LOG R REPORT B ROLL BACK P PRINT                   09.04.16
```

The Rule Activity List screen includes the status of all rules that have executed, or attempted to execute, a rule for group INTRAC.

NOTE

Your screen may appear differently, depending on your computer center's activity and the rule names used within group INTRAC.

Displaying Rule Details

1 To see all of the details for the rules listed in the Rule List Activity screen, Type **DISPLAY A** in the COMMAND field and press **Enter**.

Figure 17 Rule Activity List Screen – Display Option A

```
RULE ACTIVITY  <A>  GROUP INTRAC              USERID          JOBNAME
COMMAND ===>                                                   SCROLL===> CRSR
O JOBNAME  JOBID STEPNAME            GROUP                OWNER     INV#
  I610INGS 24883 STEP4    CONTROLB   INTRAC               M66       0001
  Rule   : ARSETVAR Mission: (NOMISS) Category : (NO-CATEGORY)
  Date   : 07/12/00 Start  : 12:03    End      : 12:03
  Status : OK       Code   : 0000     Committed: 0006  Variables
  Remark :
  Report : Y
-------------------------------- end of data ----------------------------------
  I610INGS 24883 STEP5    CONTROLB   INTRAC               M66       0002
  Rule   : ARWARN01 Mission: (NOMISS) Category : (NO-CATEGORY)
  Date   : 07/12/00 Start  : 12:03    End      : 12:03
  Status : OK       Code   : 0000     Committed: 0000  Variables
  Remark :
  Report : Y
-------------------------------- end of data ----------------------------------
  I610INGS 24883 STEP6    CONTROLB   INTRAC               M66       0003
  Rule   : ARWARN04 Mission: (NOMISS) Category : (NO-CATEGORY)
  Date   : 07/12/00 Start  : 12:04    End      : 12:04
  Status : NOTOK    Code   : 0999     Committed: 0000  Variables
  Remark :
  Report : Y
OPTIONS:   V VIEW L LOG R REPORT B ROLL BACK P PRINT                 09.07.39
```

This display shows the following information:

- Whether the rule was invoked with or without a mission

  In this display, no missions were used to invoke the rule.

- How many variables were committed by this rule

- If a report exists for this rule

2 If the information is displayed beyond one screen, use the UP and DOWN commands, or press the **PF07/PF19** and **PF08/PF20** keys, respectively, to scroll forward and backward through the display.

You can also set the number of lines for which the display will scroll, using one of the following methods:

- In the SCROLL field, located in the upper right corner of the screen, specify one of the following values:

— PAGE – Scroll a full page
— HALF – Scroll half a page
— CRSR – Scroll a page starting from the line in which the cursor is positioned
— The default value in the SCROLL field can be changed for a single instance by typing one of the valid scrolling values in the COMMAND field, and then pressing either **PF07/PF19** or **PF08/PF20**.

- Type **MAX** in the COMMAND field to scroll the maximum amount for the display, either to the very top or very bottom of the display.

  To scroll to the very top of the display, after typing **MAX** in the COMMAND field, either enter the UP command or press **PF07/PF19**. This scrolls the display to the first line of the first screen of information.

  To scroll to the very bottom of the display, after typing **MAX** in the COMMAND field, either enter the DOWN command or press **PF08/PF20**. This scrolls the display to the first line of the last screen of information.

Exploring Screen Options

To explore the options shown on the bottom of the screen you must first reset the screen display to the default view, as follows:

1. Type **DISPLAY D** (for default display) in the COMMAND field and press **Enter**.

2. Place the cursor in the OPT field at the far left of the line that includes rule ARSETVAR.

3. Type **L** for Log and press **Enter** to display the Log Messages for Jobs screen, which will display several log messages relating to the same rule name, IOAINSGS. Enter the UP and DOWN commands, or press their respective **PF07/PF19** and **PF08/PF20** keys to scroll backward and forward through the log messages.

The IOA Log contains automatically generated messages that record every significant event in the life of a mission or rule. The Log includes information showing

- when the Log message was issued
- original scheduling date of a mission or date on which the rule was run
- the user ID or owner of the rule or mission
- which rule was executed
- usually, several associated log messages, as discussed above

There is a log message noting when the rule was invoked, when it ended, and when the runtime environment ended.

REPORT Option

Another option that is frequently used is the REPORT option.

1 Press **PF03/PF15** or enter the END command to return to the Rule Activity screen.

2 Place the cursor in the OPT field at far the left of the line that includes rule ARWARN01.

3 Type **R** and press **Enter**. The **CONTROL-M/Analyzer** Report Viewing screen is displayed.

Figure 18 CONTROL-M/Analyzer Report Viewing Screen

```
--------------------- CONTROL-M/Analyzer REPORT VIEWING -------------------(BA.R)
COMMAND ===>                                                  SCROLL===> CRSR
-------------------------------------------------------------------------------
               *** CONTROL-M/ANALYZER SUMMARY OF INVOCATION REPORT ***

    ***-------------------------------------------------------------***
    *** GROUP   : INTRAC                JOBNAME  : IOAINSGS /JOB 7175 ***
    *** MISSION : (NOMISS)              RULE     : ARWARN01           ***
    *** CATEGORY: (NO-CATEGORY)         OWNER    : M66                ***
    *** RUN ON  : 11/12/00 - 10:55                                    ***
    ***-------------------------------------------------------------***
*
** ENTERING BLOCK: CHECK    OF TYPE DATA
*
PRT:  TIME IS NOT BETWEEN 12 P.M. AND 6 P.M.
PRT:  YOU'RE OK FOR NOW, BUT DO NOT RELAX
==SHOUT : 'TIME IS NOT BETWEEN 12 P.M. AND 6 P.M --- RELAX ***********' To: TS0


           *** CONTROL-M/ANALYZER INVOCATION ENDED OK USERCODE=0000 ***
 ======= >>>>>>>>>>>>>>>>>>>>>    END  OF  REPORT     <<<<<<<<<<<<<<<<<<<<< =====

PRESS END TO RETURN.  LEFT AND RIGHT TO SEE MORE.                     07.28.57
```

CONTROL-M / Analyzer creates an Invocation Report for each rule that it executes. Rules can be very complex, and may include many ON blocks that perform various balancing operations on data. The report includes information to help you manage and debug a rule after it has been invoked. This includes information automatically inserted by CONTROL-M / Analyzer, as well as messages that are produced by parameters, such as DO PRINT or DO SHOUT, that you may have included in a rule definition.

NOTE

The actual screen you see may differ slightly from the example screen shown. The rule ARWARN01 produces different messages depending on the time of day that it was run.

The example report for rule ARWARN01 includes the following information:

- The ENTERING BLOCK line indicates that the ON block called CHECK was processed, as shown in the line beginning with the words “ENTERING BLOCK.”
- Two lines of information were printed, as shown in the next two lines, beginning with the word “PRT.”
- The message was sent to a TSO user, as shown at the end of the next line.
- The rule ended OK, with a user code of 0000, as shown in the last line of the report.

VIEW Option

The VIEW option enables you to check the status of variables after a rule has been invoked.

1. Press **PF03/PF15** or enter the END command to return to the Rule Activity screen.
2. Place the cursor in the OPT field, to the far left of the job named ARWARN4A.
3. Type **V** and press **Enter** to display the Variables Committed for Rule screen.

Figure 19 Variables Committed for Rule Screen

```
-------<G>---------------- VARIABLES COMMITTED FOR RULE -----------------(BA.V)
COMMAND ===>                                                   SCROLL===> CRSR
O GROUP NAME --------- VARIABLE NAME ------ GENERATION ------------------------
  INTRAC               COST03               000
   Value         : 319302.000
   User Key      :                       Run Id  :
   Creation Date : 11/12/00   11:06:06   By User : M66      Type : B
-------------------------------------------------------------------------------
  INTRAC               INVAMT03             000
   Value         : 36874.810
   User Key      :                       Run Id  :
   Creation Date : 11/12/00   11:06:06   By User : M66      Type : B
-------------------------------------------------------------------------------
  INTRAC               COST02               000
   Value         : 541873.000
   User Key      :                       Run Id  :
   Creation Date : 11/12/00   11:06:06   By User : M66      Type : B
-------------------------------------------------------------------------------
  INTRAC               INVAMT02             000
   Value         : 679025.520
   User Key      :                       Run Id  :
   Creation Date : 11/12/00   11:06:06   By User : M66      Type : B
-------------------------------------------------------------------------------
                                                                     07.34.54
```

This screen displays key information about any Database Variables that were affected by the execution of rule ARWARN4A.

In this example, the first variable listed is named COST03. It was updated with a value of 319302, and is currently generation 000, the most recent value of this variable.

NOTE

A rule always creates a variable value as generation 000. Any previous values existing for this variable then become the next older generation (001, 002, and so on).

Further down this screen are other variables that were updated by the rule. By entering UP and DOWN commands, or pressing the corresponding **PF07/PF19** and **PF08/PF20** keys, you can examine all the variables listed on this screen.

The next topic examines the Balancing Mission definition.

Balancing Mission

A CONTROL-M/Analyzer rule can be invoked directly from a job step or application program, or triggered through a balancing mission.

Use a balancing mission when you need to specify criteria for running a rule. If the criteria are met, the rule specified by the mission is scheduled for execution. Criteria may specify:

- Scheduling, including
  - Dates and times in which to execute a rule
  - Different categories of the same mission on different occasions

    For example, a daily category is scheduled every day, but a monthly category is scheduled at month end. With this facility, different rules may be scheduled for execution on these different occasions.
- Runtime conditions
- An event or condition that must exist prior to the rule execution

For example, the accounting application may require rule BALINV to be run every day to balance an invoice file. On the weekend, the balanced invoice files are updated by inclusion of several CONTROL-M/Analyzer variables containing weekly totals. Rule UPDATE performs this function.

By using the Mission Definition facility, you also add a level of operational security to the balancing task. For example, you can define a mission which groups rule BALINV and rule UPDATE together, and schedules rule BALINV to run daily and rule UPDATE to run weekly. A job can be defined that refers to this mission rather than to either one of these rules. By doing this, the UPDATE rule cannot be mistakenly run on a weekday.

This exercise demonstrates how a rule is invoked by defining and ordering such a mission.

1. Enter the Balancing Mission Definition entry panel using one of the following methods:

- If you are working in another screen other than the Balancing Mission Definition screen, enter **=BM** in the COMMAND line.
- If you are not working in the Primary Option Menu, enter **=1**. From the Primary Option Menu, enter **BM** in the COMMAND line.

The **CONTROL-M/Analyzer** Mission Definition entry panel is displayed.

Figure 20 CONTROL-M/Analyzer Mission Definition Entry Panel

```
------------ CONTROL-M/Analyzer MISSION DEFINITION - ENTRY PANEL ------------(BM)
COMMAND ===>



SPECIFY LIBRARY, MISSION NAME, CATEGORY

LIBRARY   ===> CTB.PROD.BALMIS
MISSION   ===>                      (Blank for mission selection list)
CATEGORY  ===>                      (Blank for category selection list)








 SHOW DOCUMENTATION       ===> N     (Y/N)
 AUTO-SAVE DOCUMENTATION  ===> Y     (Y/N)




USE THE COMMAND "SHPF" TO SEE PFK ASSIGNMENT                          07.38.57
```

2 If the mission library name in the LIBRARY field is not CTB.PROD.BALMIS, as shown in the example, correct it to the library name used at your site. Leave the MISSION and CATEGORY fields blank.

3 Press **Enter** to display the Mission List screen.

Figure 21 Mission List Screen

```
MISSION LIST IN   CTB.PROD.BALMIS                              ------------(BM)
COMMAND ===>                                                   SCROLL ===> CRSR
OPT  NAME ------------ VV.MM  CREATED     CHANGED     SIZE  INIT   MOD   ID
     ADMIN
     ARMISS01
 ====== >>>>>>>>>>>>>>>>    NO MORE MISSIONS IN LIBRARY    <<<<<<<<<<<<<< =====

















OPTIONS:  S SELECT   O ORDER   F FORCE   B BROWSE   D DELETE          07.41.48
```

This screen shows the list of missions defined for library BALMIS. Locate the ADMIN and ARMISS01 missions on this list.

Options of the Mission List Screen

The Mission List screen has several options that you can perform on a mission. These options are shown at the bottom of the screen. In this exercise, you will select one of the missions to view it in more detail.

1 Position the cursor in the OPT field, to the left of the mission named ADMIN.

2 Type **B** and press **Enter** to display the Category List screen for this mission.

Figure 22 Category List Screen

```
CATEGORIES OF LIB CTB.PROD.BALMIS                                BAL MIS ADMIN
COMMAND ===>                                                     SCROLL ===> CRSR
OPT  NAME ----------------- DESCRIPTION ------------------------------ BROWSE -
     DAILY                  DAILY BALANCING MISSION FOR ACCOUNTS RECEIVABLE
     WEEKLY                 END OF WEEK SUMMARY REPORT
     MONTHLY                MONTHLY REPORTING MISSION FOR A/R
 ====== >>>>>>>>>>>>>>>> NO MORE CATEGORIES IN THE MISSION <<<<<<<<<<<<<< =====

OPTIONS:  S SELECT   O ORDER   F FORCE   C COPY                        07.43.36
```

A mission can be divided into multiple parts, each with different scheduling criteria. Each of these parts are referred to as a category.

The ADMIN mission is divided into the following three categories: DAILY, WEEKLY, and MONTHLY (see Figure 23).

Figure 23 Missions, Divided into Categories

The names of missions and their categories should be descriptive of their functionality. For example, the ADMIN mission is where you look for rules and scheduling pertaining to administration. The category names DAILY, WEEKLY, and MONTHLY are missions to be run on a daily, weekly, and monthly basis, respectively.

Actions that can be performed for each category are shown on the bottom of the screen. Note that these actions are similar to the actions on the Mission List screen. This means that you can perform similar actions on the mission as a whole, or on just one category within the mission.

Viewing the Mission Definition

In this exercise you will examine the DAILY category.

1. Place the cursor in the OPT field to the left of the mission category named DAILY.

2. Type **S** and press **Enter**. The Mission Definition screen for the DAILY category is displayed.

Figure 24 The Mission Definition Screen for Category DAILY

```
-- CONTROL-M/Analyzer CATEGORY DAILY              BAL MISSION ADMIN    ---(BM.S)
COMMAND ===>                                                  SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| CATEGORY DAILY                                MISSION ADMIN                 |
| JOB      M18GO     STEP          ,            SCOPE   JOB                   |
| RULENAME BALINV    RULELIB CTB.PROD.RULES                                  |
| OWNER    M18B      GROUP   INTRAC                                          |
| DESC     DAILY BALANCING MISSION FOR ACCOUNTS RECEIVABLE                   |
| DOCMEM   ADMIN     DOCLIB                                                  |
| ========================================================================== |
| DAYS                                                          DCAL         |
|                                                                    AND/OR O |
| WDAYS   1,2,3,4,5                                             WCAL         |
| MONTHS  1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y     |
| DATES                                                                      |
| CONFCAL          SHIFT   RETRO N   MAXWAIT 00                              |
| MINIMUM          PDS                                                       |
| ========================================================================== |
| IN      CTB-ADMIN-DATA-OK    ODAT                                          |
| TIME      FROM      TO      NOT LATER THAN      PRIORITY                   |
| ========================================================================== |
| SET VARS                                                                   |
======= >>>>>>>>>> END OF BALANCING PARAMETERS FOR THIS CATEGORY <<<<<<< ======
USE THE DOC COMMAND TO SHOW/HIDE BALANCING DOCUMENTATION              07.50.40
```

General Information Parameters: Matching a Job with a Mission

The top section of the Mission List screen contains the General Information parameters. These parameters provide general information about the mission as well as information that matches a job with a mission.

In this example, the JOB field indicates that the mission must be invoked by JOB M18GO. CONTROL-M / Analyzer performs the following steps to match JOB M18GO with this mission:

1. JOB M18GO contains a step which invokes CONTROL-M / Analyzer. When it is executed, the MISSION parameter in the JCL EXEC statement instructs CONTROL-M / Analyzer to look for the ADMIN mission.

2. CONTROL-M / Analyzer looks in its Active Balancing file for the ADMIN mission. Putting a mission in the Active Balancing file is described in “Active Balancing Environment” on page 1-52.

3. CONTROL-M / Analyzer determines, from this mission definition, that it must execute rule BALINV.

Basic Scheduling Parameters

The middle section of the screen contains the Basic Scheduling parameters for the mission. The CONTROL-M / Analyzer scheduling capabilities are extensive. As shown in the example definition, this mission is scheduled for weekdays 1 through 5, and all months. For more information on Basic Scheduling parameters, see the discussion of mission definition parameters in the *CONTROL-M/Analyzer User Guide*.

Runtime Scheduling Parameters

Following the Basic Scheduling information, there are two lines containing Runtime Scheduling parameters. In this mission, you see that condition CTB-ADMIN-DATA-OK must exist at runtime, when this mission is invoked, to execute the rule.

SET VARS: Presetting a CONTROL-M/Analyzer Variable

The last line, SET VARS, presets a CONTROL-M / Analyzer variable to a value before executing the rule. The mission does not set any variable values.

3 Press **PF03/PF15** or enter the END command. The following confirmation window is displayed:

Figure 25 Balancing Mission Definition Exit Confirmation Window

```
CATEGORIES OF LIB CTB.PROD.BALMIS                                BAL MIS ADMIN
COMMAND ===>                                                     SCROLL ===> CRSR
OPT  NAME ---                +-------------------------------+ -------- BROWSE -
 F   DAILY      <-----------|   CONFIRM  Y  ODATE 111200    | TS RECEIVABLE
     WEEKLY                  +-------------------------------+
     MONTHLY                 MONTHLY REPORTING MISSION FOR A/R
 ====== >>>>>>>>>>>>>>>>> NO MORE CATEGORIES IN THE MISSION <<<<<<<<<<<<<<< =====

OPTIONS:  S SELECT   O ORDER   F FORCE   C COPY                        07.53.42
```

4 Type **Y** in the CONFIRM field and press **Enter** to return to the Category List screen.

Ordering and Forcing Missions

A mission can be placed in the Active Balancing file automatically, by a special procedure. For more information about this special procedure, see Chapter 4, "Preparing for Production—Technical" You can also manually put a mission in the Active Balancing file by "ordering" it.

The following exercise illustrates how to manually order the DAILY category of mission ADMIN to be placed in the Active Balancing file.

Two relevant options, ORDER and FORCE, are available on this screen. While they take similar actions, the criteria followed by these options are different. The ORDER option places the mission on the Active Balancing file according to its defined schedule. The FORCE option forces the mission to be placed on the Active Balancing file, regardless of its schedule. In this example, you will order the mission using the FORCE option.

1. Place the cursor in the OPT field to the left of the mission category named DAILY.

2. Type **F** and press **Enter** to force the ordering of this mission category.

   If a pop-up confirmation window appears, type **Y** to confirm the order.

   After a brief period, the **CONTROL-M/Analyzer** Mission Order Messages screen is displayed with a confirmation message for the order. This message indicates that the mission has been ordered, and it is now placed in the Active Balancing file.

Figure 26 CONTROL-M/Analyzer Mission Order Messages Screen

```
------------------- CONTROL-M/Anlayzer MISSION ORDER MESSAGES -------------(BM.O)
COMMAND ===>                                                   SCROLL===> CRSR
-------------------------------------------------------------------------------
 07.56.51 BAO501I CTBBAO STARTED
 07.56.52 BAO528I MEMBER ADMIN    ID=0001D ODATE 131200 PLACED ON ACTIVE BALANC
 07.56.52 BAO525I CTBBAO ENDED
 ====== >>>>>>>>>>>>>>>>>>      END OF MESSAGE LIST      <<<<<<<<<<<<<<<<<< =====



PRESS END TO RETURN.  LEFT AND RIGHT TO SEE MORE.                     07.56.52
```

Active Balancing Environment

Use the CONTROL-M / Analyzer Active Balancing Environment screen to check all active missions. Generally, this facility should be used to check the status of a mission, while the Rule Activity facility should be used to check the status of a rule.

Active Balancing Environment Screen

Enter the Active Balancing Environment screen using one of the following methods:

- If you are working in another screen other than the Active Balancing Environment Definition screen, enter **=BB** in the COMMAND line.
- If you are not working in the Primary Option Menu, enter **=1**. From the Primary Option Menu, enter **BB** in the COMMAND line.

The Active Balancing Environment screen is displayed.

Figure 27 Active Balancing Environment Screen

```
------------ CONTROL-M/Analyzer ACTIVE BALANCING ENVIRONMENT -----<D>--------(BB)
COMMAND ===>                                                  SCROLL===> CRSR
O RULE     ODATE  SCP OWNER    JOBNAME ---------- STATUS ----------------------
  ARWARN01 101092 JOB M66      *        ENDED OK CODE=0000
                                        RUN-0001,OK-0001,NOTOK-0000,TOLER-0000
  ARSETVAR 111200 UNS M66      N520INGS ENDED NOTOK CODE=4091
  ARWARN01 111200 UNS M66      N520INGS ENDED OK CODE=0000
  ARWARN04 111200 UNS M66      N520INGS ENDED NOTOK CODE=4091
  ARWARN4A 111200 UNS M66      N520INGS ENDED NOTOK CODE=4091
  ARSETVAR 111200 UNS M66      N52AHEL  ENDED NOTOK CODE=4091
  BALINV   131200 JOB M18B     M18GO    WAIT ACTIVATION
======= >>>>>>>>>>>>>>>>>>>>> BOTTOM OF ACTIVE RULES <<<<<<<<<<<<<<<<<<< =======









OPTIONS:  ? WHY  H HOLD  D DEL   F FREE  L LOG                        13.46.12
```

If your data center is actively testing or using CONTROL-M / Analyzer, your screen may show a great deal of information—in fact, much more information than desired at this point in the exercise. Before you examine the Active Balancing Environment screen in detail, you will use the Active Balancing Environment SHOW command. This command allows you to control the amount of information that is displayed.

SHOW Command

1 In the COMMAND line, type **SHOW** and press **Enter** to display the Active Balancing Environment screen with the SHOW window open

Figure 28 Active Balancing Environment Show Window

```
------------- CONTROL-M/ANALYZER ACTIVE BALANCING ENVIRONMENT ----<D>--------(BB)
COMMAND ===>        +-------------------------------------------------------+
O RULE     ODATE    |                PLEASE SELECT SHOW OPTION              |
  ARSETVAR 111200   |  RULE                                                 |
                    |  JOBNAME                                              |
  ARWARN01 111200   |  MISSION                                              |
  ARWARN04 111200   |  CATEGORY                                             |
  ARWARN4A 111200   |  GROUP                                                |
  ARSETVAR 111200   |                                                       |
  BALINV   131200   |   IN PROCESS     Y  | ENDED        Y  | STATE        Y  |
======= >>>>>>>>>>  |  -------------------+-----------------+--------------  |
                    |  WAIT ACTIVATION Y  | ENDED OK     Y  | HELD         Y  |
                    |  BALANCING       Y  | ENDED NOTOK  Y  | DELETED      Y  |
                    |                     | ENDED TOLER  Y  | FREED        Y  |
                    |         === SCOPE ===                                 |
                    |   SNG   STP   JOB   ALL   UNS                         |
                    |    Y     Y     Y     Y     Y                          |
                    |  COND                                                 |
                    |  IN CONDITIONS  YES Y NO Y                            |
                    |  OWNER                                                |
                    +-------------------------------------------------------+


OPTIONS:  ? WHY  H HOLD  D DEL   F FREE  L LOG                        07.58.45
```

This screen provides many choices for viewing only the information you want. For example, you can simplify the display to show only those rules and missions belonging to group INTRAC which did not end NOTOK.

2 Place the cursor in the GROUP field, and type **INTRAC**.

3 Place the cursor in the ENDED NOTOK field, and type **N**, as shown in Figure 29.

Figure 29 Setting the GROUP and ENDED NOTOK Fields in the Show Window

```
------------- CONTROL-M/ANALYZER ACTIVE BALANCING ENVIRONMENT ----<D>--------(BB)
COMMAND ===>        +-------------------------------------------------------+
O RULE     ODATE    |                PLEASE SELECT SHOW OPTION              |
  ARSETVAR 111200   |  RULE                                                 |
                    |  JOBNAME                                              |
  ARWARN01 111200   |  MISSION                                              |
  ARWARN4A 111200   |  GROUP     INTRAC                                     |
  ARSETVAR 111200   |                                                       |
  BALINV   131200   |   IN PROCESS     Y  | ENDED        Y  | STATE        Y  |
======= >>>>>>>>>>  |  -------------------+-----------------+---------------  |
                    |  WAIT ACTIVATION Y  | ENDED OK     Y  | HELD         Y  |
                    |  BALANCING       Y  | ENDED NOTOK  N  | DELETED      Y  |
                    |                     | ENDED TOLER  Y  | FREED        Y  |
                    |         === SCOPE ===                                 |
                    |   SNG   STP   JOB   ALL   UNS                         |
                    |    Y     Y     Y     Y     Y                          |
                    |  COND                                                 |
                    |  IN CONDITIONS  YES Y NO Y                            |
                    |  OWNER                                                |
                    +-------------------------------------------------------+


OPTIONS:  ? WHY  H HOLD  D DEL   F FREE  L LOG                        07.58.45
```

4 Press **Enter**. The Active Balancing Environment screen now displays only those INTRAC group jobs ending with a status other than NOTOK.

Figure 30 Results of the Sample Show Window Selections

```
-------------- CONTROL-M/ANALYZER ACTIVE BALANCING ENVIRONMENT ---<D>--------(BB)
COMMAND ===>                                                 SCROLL===> CRSR
O RULE     ODATE  SCP OWNER    JOBNAME ---------- STATUS ----------------------
  ARSETVAR 111200 UNS M66      IOAINSGS ENDED OK CODE=0000
  ARWARN01 111200 UNS M66      IOAINSGS ENDED OK CODE=0000
  ARWARN4A 111200 UNS M66      IOAINSGS ENDED OK CODE=0009
  ARWARN01 111200 JOB M66      *        WAIT ACTIVATION
  ARWARN01 111200 JOB M66      *        ENDED OK CODE=0000
                                        RUN-0001,OK-0001,NOTOK-0000,TOLER-0000
  BALINV   131200 JOB M18B     M18GO    WAIT ACTIVATION
======= >>>>>>>>>>>>>>>>>>>> BOTTOM OF ACTIVE RULES <<<<<<<<<<<<<<<<<<< =======









OPTIONS:  ? WHY  H HOLD  D DEL   F FREE  L LOG                        08.05.23
```

You have narrowed the scope of the Active Balancing Environment display. You are now able to monitor the status of the active balancing missions that you selected. In the STATUS field for each mission, you can see that these mission requests are either awaiting execution, being executed, or have recently completed execution.

In this example, you see two status types:

- The result of missions that were invoked

  Missions either end with a status of OK or NOTOK. In this example, missions that ended had a status of OK because you previously excluded all rules that ended NOTOK. Whether a rule ends OK or NOTOK depends on the user's definition criteria for a given mission.

- Missions that have a WAIT ACTIVATION status

  WAIT ACTIVATION indicates missions waiting for a job to be run that specifies that mission's name.

On a color monitor, you are able to see how the Management by Color approach helps you separate the various tasks by result.

WHY Option

In an earlier exercise, you ordered a mission called ADMIN, which contained the rule called BALINV. Your next task will be to find out why this mission is waiting to be activated, and has not yet executed.

1. Position the cursor in the OPT field to the left of the rule named BALINV.

2. Type **?** and press **Enter** to display the WHY List screen.

Figure 31 WHY List Screen

```
------------------------- ADMIN    IS WAITING FOR: ----------------------(BB.?)
COMMAND ==>                                                     SCROLL==> CRSR

    CONDITION        CTB-ADMIN-DATA-OK       ODATE    1312
======= >>>>>>>>>>>>>>>>>      END OF ''WHY'' LIST     <<<<<<<<<<<<<<<<<< ======

PRESS END PFK TO RETURN TO STATUS SCREEN                              08.09.54
```

The WHY List screen informs you that the ADMIN mission is waiting for prerequisite condition CTB-ADMIN-DATA-OK to be satisfied. For further information about prerequisite conditions and how they are used, see the IOA environment discussion in the *CONTROL-M/Analyzer User Guide*.

3 Press **PF03/PF15** or enter the END command to return to the Active Balancing Environment screen to explore other options.

DISPLAY Command

There is other information available on the Active Balancing Environment screen that can help explain the status of a mission before, during, and after its invocation. The DISPLAY command can be used to change the DISPLAY option in order to display all information about each mission.

1 In the COMMAND field, type **DISPLAY A** and press **Enter** to show the DISPLAY A option of the Active Balancing Environment screen.

Figure 32 Active Balancing Environment Screen – Display A

```
------------- CONTROL-M/Analyzer ACTIVE BALANCING ENVIRONMENT ---<<A>--------(BB)
COMMAND ===>                                                   SCROLL===> CRSR
O RULE     ODATE  SCP OWNER    JOBNAME ---------- STATUS ----------------------
  ARSETVAR 101092 UNS M66      IOAINSGS ENDED OK CODE=0000
    GROUP:    INTRAC                   RUN-0001,OK-0001,NOTOK-0000,TOLER-0000
    CATEGORY: (NO-CATEGORY)
    MISSION:  (NOMISS)  DESC:
    JOB STEPNAME:
    PROC STEPNAME:  STEP4

  ARWARN4A 111200 UNS M66      IOAINGS ENDED OK CODE=0009
    GROUP:    INTRAC
    CATEGORY: (NO-CATEGORY)
    MISSION:  (NOMISS)  DESC:
    JOB STEPNAME:
    PROC STEPNAME:  STEP7

  ARWARN01 131200 JOB M66      *       WAIT ACTIVATION
    GROUP:    INTRAC
    CATEGORY: DAILY
    MISSION:  ARMISS01  DESC: SHOUT WAR
   NING IF DIVISIONAL AR DATA NOT IN BY
   JOB STEPAME:
OPTIONS:  ? WHY  H HOLD  D DEL   F FREE  L LOG                        08.14.39
```

2 Press **PF08/PF20** or enter the DOWN command to scroll forward until you see mission ARMISS01.

With this detailed display, you can determine which mission category was ordered. This information is important when you are testing the application to determine if you have correctly set up its parameters. It is also a useful monitoring function for operations staff to keep track of CONTROL-M / Analyzer activities.

Other information provided with this display option includes the mission description, the job step name, and the procedure step name.

This concludes the tour of the Active Balancing Environment. Next, you will examine the IOA Conditions / Resources screen, and see how it can help modify job flow.

IOA Conditions/Resources Screen

All INCONTROL products use a common facility to control the flow of jobs. The facility, called the IOA Conditions / Resources screen, also provides a way for INCONTROL products to communicate with each other.

You will now explore this function, and see how to integrate CONTROL-M / Analyzer with other INCONTROL functions.

Enter the IOA Conditions / Resources screen using one of the following methods:

- If you are working in a screen other than the IOA Conditions / Resources screen, enter **=4** in the COMMAND line.
- If you are working in a screen other than the Primary Option Menu, enter **=1**. Then, from the Primary Option Menu, enter **4** in the COMMAND line.

The IOA Conditions / Resources screen is displayed.

Figure 33 IOA Conditions/Resources Screen

```
------------------------- IOA CONDITIONS/RESOURCES ------------------------(4)
COMMAND ===>                                                 SCROLL ===> CRSR
PREFIX  ===>                COND Y CONTROL Y RES Y STAT Y    DATE 1401 - 1701
OPT TYPE      CONDITION/RESOURCE    IOAID  USE QUANTITY  MAX  *P   RBA    DATE
    RESOURCE  RES-1                        B    0004    0004
    RESOURCE  RES-2                        B    0004    0004
    RESOURCE  RIQ-MALT28-DC-HLD            B    0010    0010
    RESOURCE  RRR1                         B    0002    0002
    RESOURCE  TAPE                         B    0010    0010
    CONTROL   MYCONTROL                    E                    (00000)
    CONTROL   MYCONTROL9                   E                    (00000)
    CONTROL   MYCONTROL6                   E                    (00000)
    COND      MORD-A-OUT                                                1401
    COND      MORD-B-OUT                                                1401
    COND      $LONG-CONDITION-TEST-QM00018-2                            1401
    COND      COND1                                                     1401
    COND      COND2                                                     1401
    COND      ICH-NI-SAN                                                1501
    COND      E05-MANUAL                                                1501
    COND      E05-READ1-END                                             1501
    COND      E05-WRITE-END                                             1501
    COND      K15JOBC-CAN-START                                         1701
    COND      AAAA                                                      STAT
OPTIONS:  D DELETE  C CHANGE             COMMANDS: ADD                11.54.07
```

NOTE

The number of items you see on your IOA Conditions / Resources screen depends on the options that were chosen at your site.

Viewing Relevant Conditions

You can enter a prefix to limit the display to specific condition codes. When specifying a prefix, only those conditions with names that begin with that prefix are displayed. The following example illustrates this process.

1 In the PREFIX field, type **CTBDEMO**.

2 In the DATE field, type **1010 1010** and press **Enter**. The screen now displays only those conditions with a CTBDEMO prefix and within a date range of 1010 - 1010.

Figure 34 Viewing Relevant Conditions on the IOA/Resources Screen

```
------------------------- IOA CONDITIONS/RESOURCES ------------------------(4)
COMMAND ===>                                                 SCROLL ===> CRSR
PREFIX  ===> CTBDEMO        COND Y CONTROL Y RES Y STAT Y      DATE 1010 - 1010
OPT TYPE      CONDITION/RESOURCE    IOAID  USE QUANTITY  MAX  *P   RBA    DATE
    COND      CTBDEMO-ARDATA-NOTIN                                        1010
    COND      CTBDEMO-PROD-ENDED                                          1010
    COND      CTBDEMO-CICS-PROD-UP                                        1010
    COND      CTBDEMO-AR-ENDED-OK                                         1010
    COND      CTBDEMO-START-BACKUP                                        1010
======== >>>>>>>>>>>>>>>> B O T T O M   O F   L I S T <<<<<<<<<<<<<<<< ========

OPTIONS:  D DELETE  C CHANGE            COMMANDS: ADD                 08.26.15
```

3 Press **PF07/PF19** or enter the UP command to scroll backward to the top of the list.

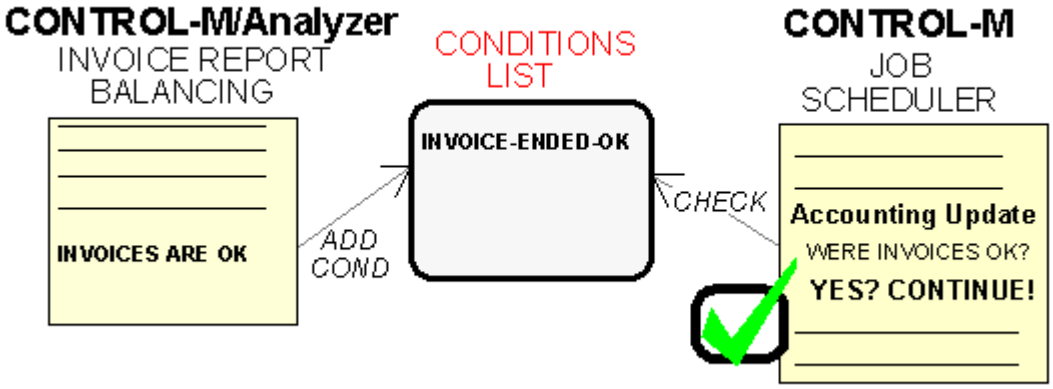
Adding Conditions

An IOA condition can be used as a prerequisite for executing a job. For example, you may not want to update the accounts receivable master file until the invoice report is balanced. You can add such a condition to a CONTROL-M / Analyzer balancing rule. The rule would then not allow the accounts receivable master file be updated until the balancing for the invoice report is OK, or within tolerable levels.

You may name a condition anything you like. However, it is recommended that you use a logical name so the operations staff can monitor events and easily understand what is happening, based on a condition name.

In the following example, the condition named INVOICE-ENDED-OK is called. If the invoice report balancing rule ends with everything in balance and OK, the rule will add the INVOICE-ENDED-OK condition with an ADD COND action.

Figure 35 Conditions as Prerequisites for Executing Jobs



When you add a condition, it appears in the IOA Conditions/Resources screen. The CONTROL-M job that executes the Accounting Update step checks the IOA Conditions/Resources list to determine if the condition exists. Before the rule can be executed, not only must the condition exist, but it also must have the correct date associated with it.

Conditions are powerful tools for controlling job flow and determining which jobs will be executed, based on the conditions found in the balancing exercise.

Conditions may be added by other INCONTROL products, or they may be added manually. The following exercise explains how to manually add conditions.

Manually Adding Conditions

Assume that in an organization the operations staff waits for an external tape containing divisional data before starting a job. Once the tape arrives, a condition is added to the Conditions/Resources list.

1 In the COMMAND line, type **ADD COND** and press **Enter**.

A window is displayed that enables you to enter the name and date of a new condition.

Figure 36 ADD COND Window

```
-------------------------- IOA CONDITIONS/RESOURCES ------------------------(4)
COMMAN +-------------------------------------------------------+  L ===> CR-R
PREFI| |     PLEA-E FILL IN COND NAME, DATE AND PRE-- ENTER      |  1010 - 1011
OPT TY |                                                        |  BA    DATE
    CO |     NAME ===>                       DDMM ===>          |        1010
    CO |                                                        |        1010
    CO +-------------------------------------------------------+        1010
    COND      CTBDEMO-AR-ENDED-OK                                        1010
    COND      CTBDEMO-START-BACKUP                                       1010
======== >>>>>>>>>>>>>>>> B O T T O M   O F   L I S T <<<<<<<<<<<<<<<< ========

OPTIONS:  D DELETE  C CHANGE            COMMANDS: ADD                13.03.48
```

2 In the NAME field of the window that is opened, type **CTBDEMO-DATA-ARRIVED**. The cursor moves to the DDMM field.

NOTE

If you had entered a name with fewer than 20 characters, you would press **Tab** to move to the DDMM field.

3 In the DDMM field, type **1010**.

Figure 37 Name and Date for a New Condition

```
-------------------------- IOA CONDITIONS/RESOURCES ------------------------(4)
COMMAN +-------------------------------------------------------+  L ===> CR-R
PREFIX |     PLEA-E FILL IN COND NAME, DATE AND PRE-- ENTER      |  1010 - 1011
OPT TY |                                                         |  BA    DATE
    CO |     NAME ===> CTBDEMO-DATA-ARRIVED  DDMM ===> 1010      |        1010
    CO |                                                         |        1010
    CO +-------------------------------------------------------+        1010
    COND      CTBDEMO-AR-ENDED-OK                                        1010
    COND      CTBDEMO-START-BACKUP                                       1010
======== >>>>>>>>>>>>>>>>> B O T T O M   O F   L I S T <<<<<<<<<<<<<<<< ========

OPTIONS:  D DELETE  C CHANGE            COMMANDS: ADD                13.03.48
```

4 Press **Enter** to display the updated list of conditions with a message at the top of the screen.

Figure 38 Updated List of Conditions

```
CTM587I COND CTBDEMO-DATA-ARRIVED ODATE 1010 ALREADY EXISTS     ------------(4)
COMMAND ===>                                                  SCROLL ===> CRSR
PREFIX  ===> CTBDEMO        COND Y CONTROL Y RES Y STAT Y     DATE 1010 - 1011
OPT TYPE      CONDITION/RESOURCE    IOAID  USE QUANTITY  MAX  *P   RBA    DATE
    COND      CTBDEMO-DATA-ARRIVED                                        1010
    COND      CTBDEMO-ARDATA-NOTIN                                        1010
    COND      CTBDEMO-PROD-ENDED                                          1010
    COND      CTBDEMO-CICS-PROD-UP                                        1010
    COND      CTBDEMO-AR-ENDED-OK                                         1010
    COND      CTBDEMO-START-BACKUP                                        1010
======== >>>>>>>>>>>>>>>>> B O T T O M   O F   L I S T <<<<<<<<<<<<<<<< ========

OPTIONS:  D DELETE  C CHANGE            COMMANDS: ADD                13.18.53
```

The new condition is now displayed on the list.

If you do not see this new condition displayed, continue to press **PF08/PF20**, or enter the DOWN command until you have scrolled to the bottom of the display.

If another user has already entered this condition, CONTROL-M / Analyzer issues a warning message indicating that the condition already exists. The preceding example includes such a warning message. You may continue with the tutorial without entering this condition.

This concludes the tour of the Conditions / Resources facility of CONTROL-M / Analyzer. The last online function that you will explore is the IOA Log facility, which contains an audit trail of INCONTROL product events.

IOA Log

INCONTROL products share a common audit trail, which is stored in the IOA Log. This common database is a major advantage if you are trying to track problems that may occur.

Often, a job or an application will use several of the INCONTROL products during its life cycle in the system. For example, a job may be scheduled by CONTROL-M. The job may start by executing a CONTROL-M / Analyzer rule to balance a file. Further down the job stream, a report may be sent to CONTROL-D, and a user may be notified.

When something unexpected happens during processing, you can look in the IOA Log to determine where the problem may have occurred. Depending on your security authorization, you may wish to explore the audit trail of specific events.

The IOA Log facility provides complete flexibility in choosing what you want to see or hide.

1 Enter the IOA Log screen using one of the following methods:

- If you are working in a screen other than the IOA Log screen, enter **=5** in the COMMAND line.
- If you are working in a screen other than the Primary Option Menu, enter **=1**. Then, from the Primary Option Menu, enter **5** in the COMMAND line.

The IOA Log screen is displayed.

Figure 39 IOA Log Screen

```
FILTER:            ---------------- IOA LOG  -------------------------------(5)
COMMAND ===>                                                     SCROLL===> CRSR
SHOW LIMIT ON ==>                                            DATE 150101 - 180101
DATE   TIME   ODATE  USERID   CODE    ------ M E S S A G E --------------------
180101 130027 180101 N98A     SPY254I JOB JOB1 N98JOB /25398 OID=001KW SCANNED
180101 130027 180101 N98A     SEL256I JOB JOB1 N98JOB /25398 OID=001KW JOB
                                      RELEASED CONTROL TEST1 MODE E
180101 130027 180101 N98A     SEL208I JOB JOB1 N98JOB /25398 OID=001KW ENDED
                                      "OK"
180101 130027 180101 N98A     SEL203I JOB JOB1 OID=001KX ELIGIBLE FOR RUN
180101 130028 180101 N98A     SUB133I JOB JOB1 N98JOB /25399 OID=001KX
                                      SUBMITTED FROM LIBRARY (P) N98.LIB.JCL
180101 130030 180101 N98A     SPY28GI JOB JOB1 N98JOB /25399 OID=001KX TAPE
                                      DRIVE UNITS USED=00 00
180101 130030 180101 N98A     SPY281I JOB JOB1 N98JOB /25399 OID=001KX START
                                      01018.1300 STOP 01018.1300 CPU 0MIN
                                      00.02SEC SRB 0MIN 00.00SEC 0.00 2AOS35
180101 130030 180101 N98A     SPY254I JOB JOB1 N98JOB /25399 OID=001KX SCANNED
180101 130031 180101 N98A     SEL256I JOB JOB1 N98JOB /25399 OID=001KX JOB
                                      RELEASED CONTROL TEST1 MODE E
180101 130031 180101 N98A     SEL208I JOB JOB1 N98JOB /25399 OID=001KX ENDED
                                      "OK"
======== >>>>>>>>>>>>>>>>>   NO  MORE  LOG  MESSAGES    <<<<<<<<<<<<<<<<< =======
CMDS: SHOW, GROUP, CATEGORY, SHPF                                     13.52.33
```

The number of messages you see, and for which INCONTROL products the messages are generated, depends on the options that were chosen for your site.

The IOA Log screen includes the following information:

Table 3 Fields of the IOA Log Screen

Field	Description
DATE	The date on which the Log message was issued
TIME	The time at which the Log message was issued
ODATE	Original scheduling date of the mission
USERID	User ID, or owner, of the mission issuing the message, or of the user writing to the log
CODE	INCONTROL message code
MESSAGE	The INCONTROL message If the message is longer than the space available on the screen, the message is split and continues on the following line. The message shows every significant event between the time the CONTROL-M / Analyzer environment is invoked until the Runtime Environment is ended.

SHOW Command

To make the IOA Log more useful for a given task, you may want to look at only part of the IOA Log screen information. You can show fewer fields with the SHOW command, as you did in previous functions.

1 In the COMMAND line, type **SHOW** and press **Enter**.

The IOA Log show screen is displayed with options that allow you to define which fields you want to display in the IOA Log screen.

Figure 40 IOA Log Show Screen

```
FILTER:         +-------------------- IOA LOG SHOW SCREEN -----------------(5)+
COMMAND ===>    |  FILTER            SAVE   (Y/N)  DESC:                      |
SHOW LIMIT 0    |  CM     : D JOB M JOB   SHOUT USER GENERAL D INT M INT STAT |
DATE   TIME     |             Y    Y        Y     Y     Y      N     N    N   |
180101 13002    |  CO+CMEM: GENERAL  SHOUT JOBS                               |
180101 13002    |             Y        Y    Y                                 |
                |  CD+CV  : SBSYS REP MIS SHOUT USER GENERAL DAILY MONIT STAT |
180101 13002    |             Y    Y   Y    Y     Y     Y      N     N    N   |
                |  CB     : RUNTIME SHOUT DAILY GENERAL STATISTICS            |
180101 13002    |             Y       Y     Y      Y        Y                |
180101 13002    |  CT     : GENERAL SHOUT      REAL-TIME UTILITIES            |
                |             Y       Y            Y        Y                |
180101 13003    |  CODE                                                       |
                |  URGENCY:      REGULAR Y   URGENT Y   VERY-URGENT Y         |
180101 13003    |  TASK TYPE     CM: JOB CYC EMR STC CST EST ECJ ECS WRN GRP  |
                |                     Y   Y   Y   Y   Y   Y   Y   Y   Y   Y   |
                |                CD: REP PRT BKP/MIG RST EMR NOEMR CYC NOCYC  |
180101 13003    |                     Y   Y     Y     Y   Y    Y    Y    Y    |
180101 13003    |  USERID                                                     |
                |  MEM/MIS                                                    |
180101 13003    |  JOBNAME                                                    |
                |  CATEGORY                                                   |
======== >>>    |  GROUP                                                      |
CMDS: SHOW,     +-------------------------------------------------------------+
```

NOTE

The screen may appear slightly different from this example, depending upon which INCONTROL products have been installed at your site.

The options in the top half of the screen allow you to choose INCONTROL products for which you want to view messages, and which category of messages you want to see displayed. Any combination of these options is valid.

In the bottom of the screen, you can select to view log activity by USERID, MISSION, JOB NAME, CATEGORY, or GROUP.

2 In the USERID field, type your user ID.

3 In the GROUP field, type **INTRAC**, and press **Enter**.

The Log now displays only those messages for your user ID and the group INTRAC. Had there been many more users in your environment, this would greatly simplify your search for messages pertaining only to your user ID.

This concludes the tour of the CONTROL-M / Analyzer IOA Log facility. In addition, you have completed the tour of the CONTROL-M / Analyzer online functions.

In the next chapter you will use a case study to develop your skills in understanding and writing CONTROL-M / Analyzer rules.

4 In the COMMAND line, type **=1** to return to the Primary Option menu.

Chapter 2

Example Problems and Solutions

This chapter includes the following topics:

Overview 68
The Example Company, INTRAC 68
INTRAC's Problems—General 71
Sources of Data—Reports and Files 72
Basic Problems and Solutions. 74
Problem 1: The Daily Invoice File — Warning 74
Problem 2: Check Payroll Totals 90
Problem 3: Check Sales Commissions 103
Advanced Problems and Solutions 108
Problem 4: Balance Divisional Accounting Files 108
Problem 5: Check Weekly Database Amounts 124
Problem 6: Balance Monthly Commissions 132
CONTROL-M / Analyzer Capabilities Summary 145

Overview

This chapter presents a set of control and balancing problems and their solutions using the example company INTRAC, which you were introduced to in the tutorial in Chapter 1. The problems are divided into Basic and Advanced. A summary of the CONTROL-M / Analyzer capabilities described in these example problems and solutions appears at the end of the chapter.

While each problem and solution may be examined individually, the INTRAC case study builds on the knowledge that you gain in working through each problem in the sequence provided. This is particularly true if you are a new CONTROL-M / Analyzer user.

The Example Company, INTRAC

INTRAC is a public company in the transportation industry. INTRAC has three major product divisions. Division 1 provides short term courier services for small and medium sized packages, with delivery worldwide. Division 2 handles containers and other large commercial shipments, which are usually delivered within a one to three month time frame. Division 3 provides a variety of transportation-related services, such as freight forwarding, insurance, and so on.

Until a few years ago, all production applications were run on the company's mainframe computer at Headquarters (HQ). Five years ago, the Central Information Systems Division (ISD) began distributing selected applications to various PC networks at the divisions.

Major applications and financial databases continue to run on the mainframe computer, for several reasons:

- Complexity of the corporate applications
- Implications of tight financial control required of a public company
- Tighter security on the mainframe than on networks
- Client desire for discounts and accounting statements based on their total business with INTRAC

Clients view INTRAC as a single organization, through an integrated sales force that sells INTRAC's total services.

This simple concept has made information management at INTRAC more complex. Accordingly, INTRAC HQ must:

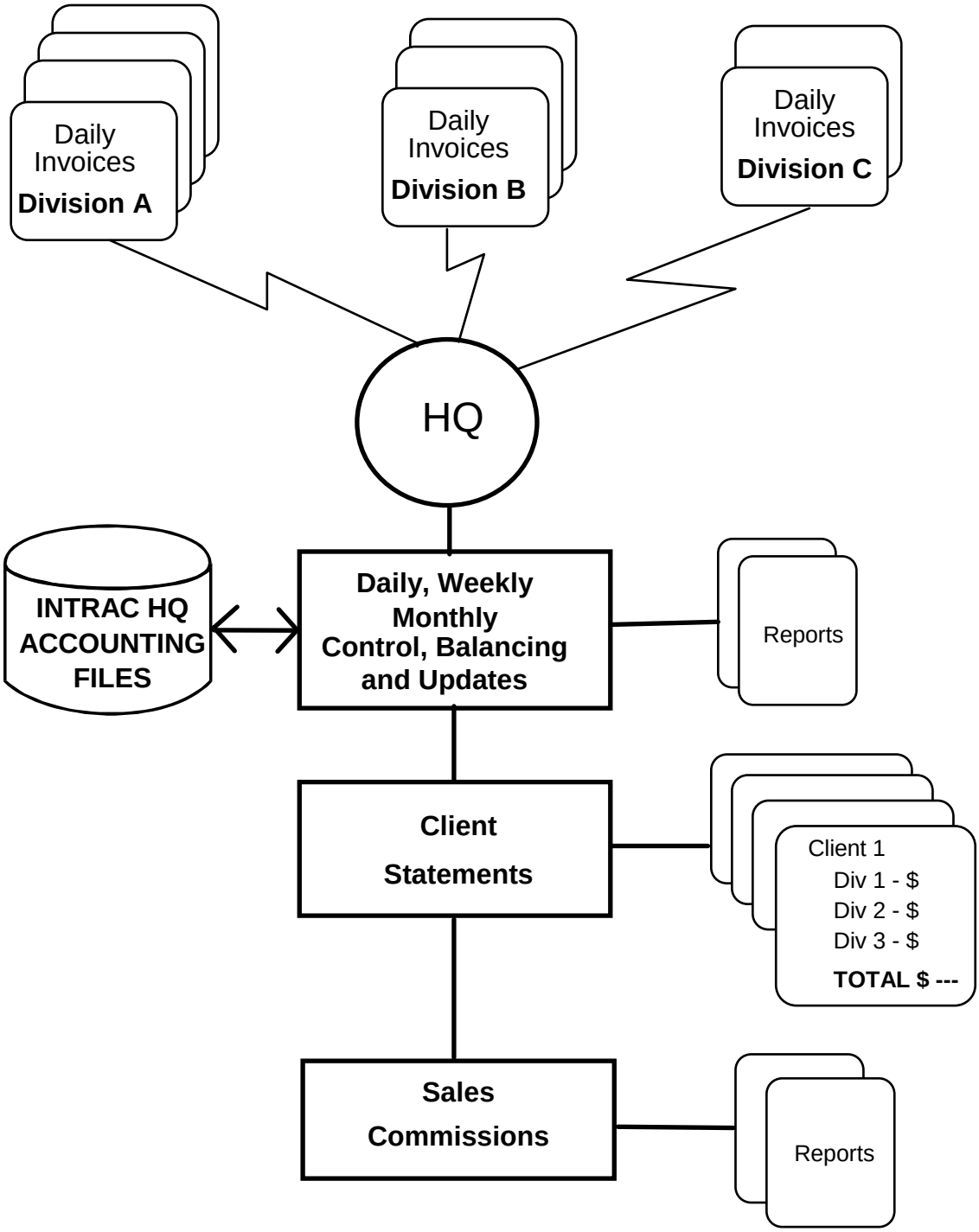
- Get transactions that are entered on divisional computers and are handled by many different people, and consolidate the information at HQ
- Thoroughly check the transactions
- Convince executive, management, and sales personnel that each has all the necessary and correct information on which to base reports and sales commissions

Each day the divisions transmit details of their daily sales results to HQ. On a daily basis, HQ updates their accounting system files. At scheduled times during the month, HQ generates daily, weekly and monthly reports. These reports are sent to dozens of locations.

At month-end, after processing all adjustments, consolidated client statements are issued, and sales commissions are calculated. A file containing the commission amount to be paid to each salesperson is sent from the sales operations department to the HQ payroll department. Additional reports are generated and sent to executives and management personnel, and to every salesperson.

Figure 41 illustrates INTRAC's work flow.

Figure 41 INTRAC Work Flow



INTRAC's Problems—General

Several times each month, there are qualitative and timeliness problems with the data sent from the divisions. Problems may be caused by transmission errors, inexperienced staff at the divisions, application programming errors, tampering with the data, missed schedules, or a host of other reasons.

As a result, management receives misleading or late reports.

Also, sales commissions are often in error, because they are based on incorrect invoice or accounts receivable information.

This has resulted in management not trusting the reports and not taking the action that they should be taking as a result of the information received. Also, sales employee morale is suffering because sometimes there are serious sales commission errors. Occasionally, the executives have gone through internal reviews and meetings with their bank, using inaccurate information.

Each new application that is distributed to the PC Network has resulted in additional staff being added at HQ for monitoring and control of the application. HQ staff levels are now frozen and management wants tighter controls and procedures to be implemented immediately.

ISD programming staff have a two year backlog of work. As a result, the ISD management cannot free up programmers to write more control programs for at least six months. Because of recent turnover, operations staff are working longer shifts, with more junior operators. This has made the production environment more error-prone.

Using this historical basis, you will work with the INTRAC information to simplify and improve operations.

Sources of Data—Reports and Files

INTRAC's sources of data includes the following two items:

- Daily Invoice file
- Sales Commission Detail report

Daily Invoice File

Each day, each division transmits the Daily Invoice file to an HQ PC. The file is sent with an accompanying control total record. At the end of each day, the three divisional files are consolidated into a single file on the PC and uploaded to the mainframe for processing. The file format is as follows:

Table 4 File Format for Daily Invoice File

Position	Description	Control/Balancing Criteria
1- 2	Division #	Must be '01', '02', '03'
3-30	Customer Name, or 'Total Invoices' for division total records	
31-38	Customer #	
39-45	Invoice #	
46-51	Invoice Date	Must be Current Month & Year
52-61	Amount Invoiced	Division 1 > $400,000, NOT-OK > $200,000, Tolerance Warn <= $200,000 OK Division 2 or Division 3 > $200,000, NOT-OK > $100,000, Tolerance Warn <= $100,000 OK
62-71	Cost of Goods Sold	
72-81	Special Charges	

In the list of control and balancing criteria, “Tolerance Warn” means that the invoice data is probably OK, but the amount is out of the ordinary. The user wants to see amounts such as these included as part of a report, and wants them marked “Warning,” to alert them for further checking. Such items will not hold up production processing.

NOTE

The division total records have the same format and are also found in the Daily Invoice file, following each division's detail invoice information. The division total records contain “Total Invoices” in the customer name field and the total amounts invoiced for each “Amount Invoiced,” “Cost of Goods Sold,” and “Special Charges” in columns 52-61, 62-71, and 71-81, respectively. These divisional totals must equal the sum of the detail records for that division.

Sales Commission Detail Report

Figure 42 shows a sample page from another source of data for the balancing rules—the Sales Commissions Detail report. This report is prepared monthly by the sales operations department. The report contains a list of all the invoices for which sales commissions are calculated for a given month.

At the end of each list of invoices for a given salesperson is a summary of calculated information, along with other information. The final line shows the amount of commissions that salesperson is owed, according to the sales operations department.

The several rules that analyze the lines on this report are illustrated in “Basic Problems and Solutions” page 2-74 and “Advanced Problems and Solutions” on page 2-108.

Figure 42 Sales Commission Detail Report

```
----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9---
INTRAC INC.                 SALES OPERATIONS DEPARTMENT     DATE 10/10/01 TIME 23:45  PAGE  4

 S A L E S     C O M M I S S I O N S    D E T A I L           FROM 10/10/01   TO  10/10/01
|SALESPERSON   MARYJANE INKSTERR(177) | INVOICE | INVOICE DATE |  AMOUNT    | STATUS
|                                     | B7003003| 02/02/00     |  11800.00  |         |
|                                     | B7003420| 03/03/00     |   2700.00  |         |
|                                     | B7004009| 04/04/00     |   1500.00  |         |
|                                     | B7018700| 05/05/00     |    500.00  |         |
+-------------------------------------------------------------------------------------------
| TOTAL SALES - THIS MONTH                                          16500.00 (PAID IN FULL}
| ADJUSTMENTS                                                           0.00
| YTD SALES (INCLUDING THIS MONTH)                                 999879.00
| YTD COMMISSIONS PAID                                                  0.00
| YTD COMMISSIONS OWED                                              39995.16
| COMMISSION PAYROLL AMOUNT                                         39995.16
|-------------------------------------------------------------------------------------------
```

Basic Problems and Solutions

This section discusses the following two problems and their solutions:

- Problem 1: The Daily Invoice File—Warning
- Problem 2: Check Payroll Totals
- Problem 3: Check Sales Commissions

Problem 1: The Daily Invoice File — Warning

In this problem, the Daily Invoice file arrives too late to check or correct errors.

To provide management with accurate and timely reports, the ISD staff needs time to run the reports and check with divisional users to identify and correct any error conditions. Often, the operations staff attempts to run the invoice processing job during the early afternoon, only to find that the job fails to run because one of the files was not sent from a division, or was received in error.

The operations staff often gets involved in other situations and then forgets, or doesn't have time to follow up on, this problem until it is too late to make that day's schedule.

ISD management becomes aware that there is a problem when the executives and divisional managers phone to complain that they haven't received their reports, or that their reports are in error.

Solution 1 — Overview

The consolidated daily invoice file must be available for processing on the mainframe by noon of each day. If it is not available, management must be informed automatically.

ISD has decided that if this problem continues, notice thereof must be escalated throughout the afternoon to make higher levels of management aware of the situation.

Solution — Details

A CONTROL-M/Analyzer rule is invoked hourly, starting at noon. CONTROL-M or another automated job scheduler will invoke the rule hourly, as long as the data has not arrived. The rule called ARWARN01 is run to notify various levels of users and management by sending a message to their TSO user IDs.

If the computer system time is between 12-1 P.M., a message is shouted to the operations manager on his or her user ID.

If the computer system time is between 1-3 P.M. and the condition still exists, the manager of ISD is informed. If the system time is 3 P.M. or later, with the same problem, the ISD vice president is informed.

Creating the ARWARNO1 Rule

1 From the IOA Primary Option Menu, enter the Rule Definition facility by typing **BR** in the COMMAND field and pressing **Enter**. The Rule Definition entry panel is displayed:

Figure 43 CONTROL-M/Analyzer Rule Definition Entry Panel

```
--------------- CONTROL-M/Analyzer RULE DEFINITION - ENTRY PANEL ------------(BR)
COMMAND ===>



SPECIFY LIBRARY, RULE

   LIBRARY ===> CTB.PROD.RULES
   RULE    ===> xxxxxxxx            (Blank for rule selection list)



USE THE COMMAND "SHPF" TO SEE PFK ASSIGNMENT                          13.41.48
```

2 If the library name in the LIBRARY field in this example is different from your CONTROL-M / Analyzer RULES library name, change it to the one shown in this example.

3 Enter a rule name using up to eight characters to provide a unique name to the rule. For example, this might be your own last name, user ID, or a name provided to you by your system administrator.

4 Press **Enter**. A window is displayed for defining the new rule parameters.

Figure 44 New Rule Definition Screen

```
CTB303E FILL IN THE REQUIRED FIELD                                RULE :
COMMAND ===>                                                      SCROLL===> CRSR
+-------------------------------------------------------------------------------+
| OWNER                              GROUP                                      |
| UPDATED 13/12/00  -  13:43:34      BY M86                                     |
| DESC                                                                          |
| OPTIONS                                                                       |
| ============================================================================ |
| EXECUTE          UPON                                                      C  |
| ON                                                                            |
======= >>>>>>>>>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<<<<<< =====

PLEASE FILL IN RULE DEFINITION.                                        13.43.34
```

5 Begin entering the definition for the new rule by filling in the descriptive information for the following fields:

Table 5 Fields to Define a New Rule

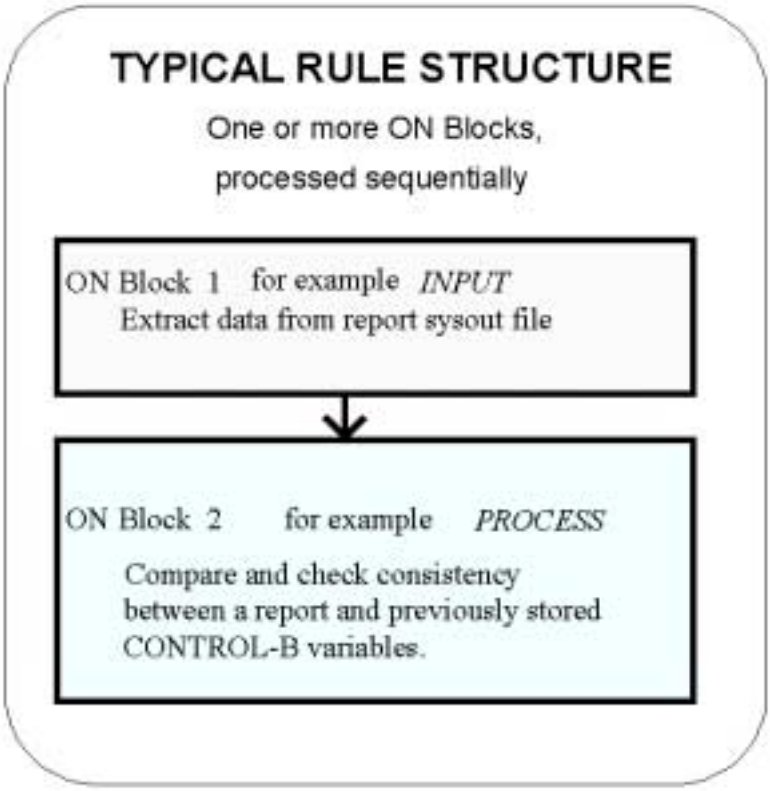
In field:	Enter the following:
OWNER	Fill in your user ID
GROUP	INTRAC
DESC	SHOUT TO USERS IF DIVISIONAL AR DATA NOT IN

6 Before you enter the other rule definition parameters, first examine the basic rule structure. A rule generally performs four types of operations:

- performs one-time actions, such as printing report headings or setting values of variables to be used later in the rule
- specifies data sources to search for and data items to be validated, either
  - from outside CONTROL-M / Analyzer (for example, file, report), or
  - from inside CONTROL-M / Analyzer (for example, CONTROL-M / Analyzer database)
- Analyzes or balances the data
- Takes action for out-of-balance and in-balance situations

These operations are usually performed in separate groupings of specifications called ON blocks. Every rule is made up of a series of one or more ON blocks. A simple rule may have just one ON block. A complex rule may have many ON blocks, as shown in Figure 45.

Figure 45 Typical Rule Structure



This first rule is very simple. The only data item with which you need concern yourself is available to you within CONTROL-M / Analyzer—the computer system time. All you need to do is check the time and issue various warnings to various users, based on how late in the day it is.

Usually, two types of ON blocks are required in a rule—Input blocks, to extract data from a file or report, and ON DATA blocks, to perform comparisons and balancing actions on the extracted data. In this rule, you will use only an ON DATA block, as illustrated in Figure 46.

You can enter and update rules using a standard system editor if you are familiar with CONTROL-M / Analyzer rule syntax. Because you are a beginning user, continue using this “fill-in-the-blanks” approach to specify and verify the rule parameters.

Figure 46 ON DATA Block Structure

```
ON DATA BLOCK STRUCTURE

EXECUTE block name UPON execution criteria
    ON DATA
        LABEL:  label
        ALWAYS
                DO action
                DO action
        LABEL:  label
        IF      expression
                DO action
                DO action
        ELSE
                DO action
                DO action
```

When you work with CONTROL-M / Analyzer variables, instead of external data such as reports or files, the ON block is structured as shown in Figure 46. In that ON block structure

- the balancing instructions begin with an EXECUTE statement

NOTE

All ON DATA blocks begin with an EXECUTE statement.

- the first LABEL is defined with an ALWAYS statement, indicating that the DO actions that follow it will always be performed on the data
- the second LABEL is defined with an IF statement, indicating that the DO actions that follow it are performed if certain criteria are met, otherwise, the DO actions following the ELSE statement are performed

This gives you the capability of taking different balancing actions on the data, depending on your needs.

Creating an ON block

You will now define the first, and only, ON block for this rule.

1. Place the cursor in the EXECUTE field.
2. Type **CHECK**. This name is descriptive of the ON block's function.

   The name of an ON block may be any eight characters, and is unique within the rule.
3. Place the cursor in the ON field.
4. Type **DATA** and press **Enter**.

The value, DATA, indicates that this is an ON DATA block type. The ON DATA block indicates that the rule is operating on data, not on reports or other external sources of information.

CONTROL-M / Analyzer now creates a blank field and prompts you to specify either **ALWAYS** or **IF** as the value.

Figure 47 Setting the ON Block Type

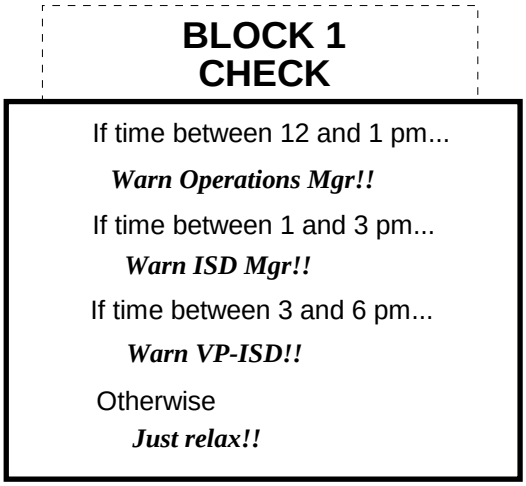
```
CTB303E FILL IN THE REQUIRED FIELD                              RULE :
COMMAND ===>                                                    SCROLL===> CRSR
+-----------------------------------------------------------------------------+
| OWNER   M86                      GROUP INTRAC                               |
| UPDATED 13/12/00  -  13:43:34    BY M86                                     |
| DESC    SHOUT TO USERS IF DIVISIONAL AR DATA NOT IN                         |
| OPTIONS                                                                     |
| ========================================================================== |
| EXECUTE CHECK    UPON                                                     C  |
| ON DATA                                                                     |
| IF           (ENTER "ALWAYS", "IF")                                         |
======= >>>>>>>>>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<<<<<< =====
PLEASE FILL IN RULE DEFINITION.                                        13.50.56
```

5. Type **IF** and press **Enter**.

Defining the ON DATA Block

You will now define the ON DATA block to do the following:

Figure 48 Flow of the ON DATA Block to Create



As illustrated in Figure 48, in the first IF condition you've just entered you will define the ON block to check for a time range between 12 and 1 P.M., and then perform the action "Warn Operations Manager!!"

To check the time, you need to access the computer system time with a system variable called SYSTIME. It has the format *hhmmss*, meaning a 6-digit number containing hours (*hh*), minutes (*mm*), and seconds (*ss*). The system clock is a 24-hour clock. This means that 1 o'clock in the afternoon, for example, appears as 130000 when read by the SYSTIME variable.

For more information on CONTROL-M / Analyzer system variables, refer to the *CONTROL-M/Analyzer for z/OS User Guide.*

With this information, you can now write the first IF condition.

1. To the right of the word IF, type **%%SYSTIME >= 120000 AND %%SYSTIME < 130000**. This means, "Is the time between 12 P.M. and 1 P.M.?"

2. Press **Enter**.

Figure 49 Definition of the First IF Condition

```
        LIBRARY : CTB.PROD.RULES                                RULE :
COMMAND ===>                                                    SCROLL===> CRSR
+-------------------------------------------------------------------------------+
| OWNER   M86                      GROUP INTRAC                                 |
| UPDATED 13/12/00  -  13:43:34    BY M86                                       |
| DESC    SHOUT TO USERS IF DIVISIONAL AR DATA NOT IN                           |
| OPTIONS                                                                       |
| ============================================================================ |
| EXECUTE CHECK    UPON                                                     C   |
| ON DATA                                                                       |
| LABEL:                                                                        |
| IF       %%SYSTIME >= 120000 AND %%SYSTIME < 13000                        C   |
|   DO                                                                          |
|              (ENTER "ALWAYS" OR "IF")                                         |
| ============================================================================ |
| EXECUTE          UPON                                                     C   |
| ON                                                                            |
======= >>>>>>>>>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<<<<<< =====




PLEASE FILL IN RULE DEFINITION.                                        13.53.51
```

Notice the character “C” at the end of the line you just typed—this is the C (Continue) field. If you want to continue this same IF expression onto another line, you would enter a Y (for YES) in this field. An IF expression sometimes may be long and complex, requiring several lines.

3 Because you do not need another line for this very simple IF parameter, place the cursor in the DO field on the next line.

Defining the Actions to Perform

You will now instruct CONTROL-M / Analyzer to do the following:

- print the message
- send a message to the operations manager
- resolve a problem if it occurs

Printing the message will provide a hard copy record that the rule was executed, and also illustrates the output that CONTROL-M / Analyzer produces to help you understand what happened during execution of the rule.

1 With the cursor in the DO field, type **PRINT** and press **Enter**. CONTROL-M / Analyzer creates a field in which to enter your print information.

2 In the new field, type **DIV. INVOICES NOT IN, PLEASE CHECK IMMEDIATELY** and press **Enter**. CONTROL-M / Analyzer creates an empty DO line below the line you’ve just completed.

Figure 50 Defining the DO Statements in an IF Condition

```
        LIBRARY : CTB.PROD.RULES                                RULE :
COMMAND ===>                                                    SCROLL===> CRSR
+-------------------------------------------------------------------------------+
| OWNER   M86                      GROUP INTRAC                                 |
| UPDATED 13/12/00  -  13:43:34    BY M86                                       |
| DESC    SHOUT TO USERS IF DIVISIONAL AR DATA NOT IN                           |
| OPTIONS                                                                       |
| ============================================================================ |
| EXECUTE CHECK    UPON                                                      C   |
| ON DATA                                                                       |
| LABEL:                                                                        |
| IF       %%SYSTIME >= 120000 AND %%SYSTIME < 13000                         C   |
|   DO PRINT    = DIV. INVOICES NOT IN, PLEASE CHECK IMMEDIATELY      F   C   |
|   DO PRINT    = ******** ACTION REQUIRED WITHIN 30 MINUTES *******  F   C   |
|   DO                                                                          |
|              (ENTER "ALWAYS", "IF" OR "ELSE")                                 |
| ============================================================================ |
| EXECUTE          UPON                                                     C   |
| ON                                                                            |
======= >>>>>>>>>>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<<<<<< =====



PLEASE FILL IN RULE DEFINITION.                                          13.55.47
```

3 In this new DO line, type **PRINT** and press **Enter**.

4 In the new field to the right of the equal sign (=), type:

********** **ACTION REQUIRED WITHIN 30 MINUTES *******

5 Press **Enter**.

You will now define the command to send (“shout”) the same message to the operations manager. For the purposes of this demonstration, you will actually send the message to yourself instead of the operations manager.

6 Place the cursor in the new DO field, type **SHOUT**, and press **Enter**.

Figure 51 DO SHOUT Definition

```
        LIBRARY : CTB.PROD.RULES                                RULE :
COMMAND ===>                                                    SCROLL===> CRSR
+-------------------------------------------------------------------------------+
| OWNER   M86                      GROUP INTRAC                                 |
| UPDATED 13/12/00  -  13:43:34    BY M86                                       |
| DESC    SHOUT TO USERS IF DIVISIONAL AR DATA NOT IN                           |
| OPTIONS                                                                       |
| ============================================================================ |
| EXECUTE CHECK    UPON                                                     C   |
| ON DATA                                                                       |
| LABEL:                                                                        |
| IF       %%SYSTIME >= 120000 AND %%SYSTIME < 13000                        C   |
|   DO PRINT    = DIV. INVOICES NOT IN, PLEASE CHECK IMMEDIATELY      F   C   |
|   DO PRINT    = ******** ACTION REQUIRED WITHIN 30 MINUTES *******  F   C   |
|   DO SHOUT    TO TSO-M86             URGENCY U                              |
|    = DIV. INVOICES NOT IN, PLEASE CHECK IMMEDIATELY *************            |
|   DO                                                                          |
|              (ENTER "ALWAYS", "IF" OR "ELSE")                                 |
| ============================================================================ |
| EXECUTE          UPON                                                     C   |
| ON                                                                            |
======= >>>>>>>>>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<<<<<< =====

PLEASE FILL IN RULE DEFINITION.                                        14.16.07
```

7 In the field that is now displayed, type your user ID, as follows:

TSO-xxxxxxxx

where *xxxxxxxx* is your TSO user ID.

8 Tab to the field labeled URGENCY and enter the urgency Code **U** (URGENT).

9 In the field to the right of the equal sign (=) on the next line, enter the following message:

DIV. INVOICES NOT IN, PLEASE CHECK IMMEDIATELY **********

To demonstrate what happens when a problem occurs, you will end the execution of this part of the rule as if there was a problem. When the operations staff reviews the log, they will see that this rule ended NOTOK.

10 Position the cursor in the next DO field and type **TERM**.

11 Press **Enter**. A DO TERMINAT action is now created.

The DO TERMINAT definition contains two fields for you to specify: =, for a result value, and COD, for a result code. In the following steps you will enter **NOTOK** as the result, and a sample code that informs the operations manager that this is the first warning. For the purposes of this demonstration, use the code **0009**.

Figure 52 DO TERMINATE Definition

```
        LIBRARY : CTB.PROD.RULES                                RULE :
COMMAND ===>                                                    SCROLL===> CRSR
+-------------------------------------------------------------------------------+
| OWNER   M86                      GROUP INTRAC                                 |
| UPDATED 13/12/00  -  13:43:34    BY M86                                       |
| DESC    SHOUT TO USERS IF DIVISIONAL AR DATA NOT IN                           |
| OPTIONS                                                                       |
| ============================================================================ |
| EXECUTE CHECK    UPON                                                     C   |
| ON DATA                                                                       |
| LABEL:                                                                        |
| IF       %%SYSTIME >= 120000 AND %%SYSTIME < 13000                        C   |
|   DO PRINT    = DIV. INVOICES NOT IN, PLEASE CHECK IMMEDIATELY      F   C   |
|   DO PRINT    = ******** ACTION REQUIRED WITHIN 30 MINUTES *******  F   C   |
|   DO SHOUT    TO TSO-M86             URGENCY U                              |
|    = DIV. INVOICES NOT IN, PLEASE CHECK IMMEDIATELY *************            |
|   DO TERMINAT = NOTOK    COD 0009                                             |
|   DO                                                                          |
|              (ENTER "ALWAYS", "IF" OR "ELSE")                                 |
| ============================================================================ |
| EXECUTE          UPON                                                     C   |
| ON                                                                            |
======= >>>>>>>>>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<<<<<< =====
PLEASE FILL IN RULE DEFINITION.                                        14.20.27
```

12 Place the cursor to the right of the equal sign (=), and type **NOTOK**.

13 Place the cursor in the COD field and type **9**.

14 Press **Enter**.

NOTE

The value **9** you just entered in the COD field is displayed with leading zeros.

Completing the Rule

You will now finish defining this rule and then test it.

Because you are simulating real conditions, you will define the following two actions which you would change when you implement the rule in production mode:

- Send all messages to yourself rather than to another user
- Send yourself a message even when the system time is not critical, just to prove to yourself that the rule is working

1 Continue entering the rule, as defined in Figure 53. Use the same procedure for entering IF statements that you used earlier by typing **IF** and then pressing **Enter**.

The word LABEL: is displayed after you enter the IF command

NOTE

Press **PF08/PF20** to scroll to the next screen when entering characters.

Figure 53 The Entire ARWARN01 Rule

```
        LIBRARY : CTB.PROD.RULES                                RULE :
COMMAND ===>                                                    SCROLL===> CRSR
+-----------------------------------------------------------------------------+
| OWNER   M86                      GROUP INTRAC                               |
| UPDATED 13/12/00  -  13:43:34    BY M86                                     |
| DESC    SHOUT TO USERS IF DIVISIONAL AR DATA NOT IN                         |
| OPTIONS                                                                     |
| ========================================================================== |
| EXECUTE CHECK    UPON                                                    C  |
| ON DATA                                                                     |
| LABEL:                                                                      |
| IF       %%SYSTIME >= 120000 AND %%SYSTIME < 13000                       C  |
|   DO PRINT    = DIV. INVOICES NOT IN, PLEASE CHECK IMMEDIATELY      F   C  |
|   DO PRINT    = ******** ACTION REQUIRED WITHIN 30 MINUTES *******  F   C  |
|   DO SHOUT    TO TSO-M86             URGENCY U                            |
|    = DIV. INVOICES NOT IN, PLEASE CHECK IMMEDIATELY *************          |
|   DO TERMINAT = NOTOK    COD 0009                                           |
|   DO                                                                        |
| LABEL:                                                                      |
| IF       %%SYSTIME >=13000 AND %%SYSTIME <150000                         C  |
|   DO PRINT    = DIV. INVOICES STILL NOT IN, URGENTS!!!              F   C  |
|   DO PRINT    = ****************** HELP NEEDED NOW **************** F   C  |
|   DO SHOUT    TO TSO-M86             URGENCY U                            |
|    = DIV. INVOICES STILL NOT IN, URGENT ACTION REQUIRED *********         |
|   DO TERMINAT = NOTOK    COD 0099                                           |
|   DO                                                                        |
| LABEL:                                                                      |
| IF       %%SYSTIME >= 150000 AND %%SYSTIME <180000                       C  |
|   DO PRINT    = DIV. INVOICES NEVER ARRIVED, EXECUTIVES WILL SUFFER F   C  |
|   DO PRINT    = ***** WHAT WILL YOU DO ABOUT IT? ****************** F   C  |
|   DO SHOUT    TO TSO-M86             URGENCY V                            |
|    = DIV. INVOICES NEVER ARRIVED, EXECUTIVES WILL SUFFER ********         |
|   DO TERMINAT = NOTOK    COD 0999                                           |
|   DO                                                                        |
| ELSE                                                                        |
|   DO PRINT    = TIME IS NOT BETWEEN 12 P.M. AND 6 P.M.              F   C  |
|   DO PRINT    = YOU'RE OK FOR NOW, BUT DO NOT RELAX                 F   C  |
|   DO SHOUT    TO TSO-M86             URGENCY R                            |
|    = TIME IS NOT BETWEEN 12 P.M. AND 6 P.M.                               |
|   DO TERMINAT = OK       COD 0000                                           |
|   DO                                                                        |
|              (ENTER "ALWAYS", "IF")                                         |
| ========================================================================== |
| EXECUTE          UPON                                                   C  |
| ON                                                                          |PLEASE
FILL IN RULE DEFINITION.                                        14.28.31
```

2 Since this is a new rule, press **PF03/PF15** or enter the END command to exit rule definition and create the rule. A confirmation window is displayed.

3 Place the cursor in the CREATE field, type **Y**, and press **Enter**.

Testing the Rule

The rule you just defined would probably be invoked by an automated operations product. For example, a job scheduler, such as the INCONTROL product, CONTROL-M, would check for the existence of the divisional file at predetermined times. If the file is not found, the rule is invoked to warn the user.

You will now simulate this situation by executing this rule and seeing if it does what you expect it to do.

In the IOA SAMPLE library there is a member named CTBRULE1. To execute the rule you just created, you need to edit CTBRULE1 and then submit a job. In order to do this, you need to know how to:

- submit a job for execution
- monitor the status of a job (using SDSF, ROSCOE, or a similar facility)
- review the results on the screen

If you are not familiar with how to perform these functions, ask your system administrator for assistance.

1 Edit the CTBRULE1 member using ISPF, ROSCOE, and so on.

Figure 54 Editing the CTBRULE1 Member

```
EDIT ---- IOA*.*.SAMPLE(CTBRULE1)  -  01.05 ------------------ COLUMNS 001 072
COMMAND ===>  SUBMIT                                           SCROLL ===> HALF
***** **************************** TOP OF DATA ********************************
000001 //ARWARN01 JOB ,GERR,CLASS=A,MSGCLASS=X,NOTIFY=M66A,REGION=4096K
000002 //STEP01   EXEC CTBTROLB,RULE=ARWARN01,GROUP-INTRAC
000003 //


***** ************************** BOTTOM OF DATA ******************************
```

Note that STEP01 has been set to invoke CONTROL-M / Analyzer, then execute rule ARWARN01 from within group INTRAC. At this time you may

- copy this JCL, then change the rule name and submit the job to run your own rule, or
- submit the above job and run the supplied rule

2 Submit the job.

As you continue working on your system and the CONTROL-M / Analyzer rule executes, you may notice a message similar to the following displayed on your screen:

```
CONTROL-B **********                                    CN (00)
CONTROL-B DIV. INVOICES NOT IN, PLEASE CHECK IMMEDIATELY *** 10.08 12:53 CN (00)
```

The exact text of the message depends on the time of day. If you do not receive a similar message on your screen, you may want to discuss this with your system administrator. However, you may continue this process even without receiving the message.

3 Analyze the output for the report

The execution of the rule also creates several output files. There are three CONTROL-M / Analyzer outputs that help you understand rule execution:

- SYSUSER – a listing containing only what you programmed, using DO PRINT commands in the rule
- SYSPRINT – a listing of the CONTROL-M / Analyzer Invocation Report which summarizes the execution of the rule
- SYSRULE – a listing of the rule

Why do you need a listing of the rule when it is executed? Anyone who is authorized to update the rule library may be able to change a rule definition at any time. By having a listing of the rule at its execution, you know exactly how the rule was defined at that given time. You can save a great deal of guesswork when diagnosing a problem, by always having the correct documentation of a rule definition to work from.

If the rule executed correctly, the three outputs appear as follows:

NOTE

You may review the job results using a tool such as SDSF.

Figure 55 SYSUSER REPORT (Executed Outside Check Hours)

```
TIME IS NOT BETWEEN 12 P.M. AND 6. P.M.
YOU'RE OK FOR NOW, BUT DO NOT RELAX
```

NOTE

The report for rule ARWARN01 will vary, depending upon what time you execute the rule.

Figure 56 SYSPRINT — Invocation Report

```
CONTROL-M/Anlayzer V6.1.00 - RUN TIME ENVIRONMENT
(C) BMC Software INC. 2001
Mission:
Rule   : ARWARN01
Group  : INTRAC

CTB10II - Compilation started
CTB110I - Compilation completed successfully
CTB111I - Highest Error Code was 0
CTB001I - CONTROL-M/ANALYZER RUNTIME ENVIRONMENT STARTED - MODE IS STAND-ALONE - SCOPE
IS
UNSCHEDULED
CTB009I - GROUP-MODE IS “ACTIVE”
CTB00AI - INVOKED RULE-ARWARN01
             *** CONTROL-M/ANALYZER SUMMARY OF INVOCATION REPORT ***

     ***-----------------------------------------------------------------***
     *** GROUP   : INTRAC                JOBNAME  : M66CTB   /JOB 1280 ***
     *** MISSION : (NOMISS)              RULE     : ARWARN01           ***
     *** CATEGORY: (NO-CATEGORY)         OWNER    : M66CTB             ***
     *** RUN ON  : 10/10/01 - 12:13                                    ***
     ***-----------------------------------------------------------------***
*
** ENTERING BLOCK: CHECK    OF TYPE DATA
*
DIV. NVOICES STILL NOT IN, URGENT!!!
******************* HELP NEEDED NOW **********************
=SHOUT : ‘DIV. INVOICES STILL NOT IN, URGENT ACTION REQUIRED********’ To: TSO-
CTB013I - ENDED “NOTOK” COMMITTED    0 VARIABLE(S) USERRC=0099
CTB002I - CONTROL-M/ANALYZER RUNTIME ENVIRONMENT ENDED

         *** CONTROL-M/ANALYZER INVOICATION ENDED NOTOK USERCODE-OO99 ***
```

NOTE

This Invocation Report was produced from rule ARWARN01, run between 1 and 3 pm. The report shows

- the ENDED status, which is OK or NOTOK, as above, depending upon what time you executed the rule
- the user code, which is 0000, or 0009, or 0099, or 0999, also depending upon the time of execution
- that the SHOUT was performed

Figure 57 SYSRULE Report

```
 *
 *** RULE:ARWARN01 from DD  : DABRULE  ***
 *
000001
000002 HEADER GROUP ‘INTRAC’ OWNER ‘M66’ DATE ‘101001’ TIME ‘155339’ CREATOR ‘M
000003 66’ DESC ‘SHOUT TO USERS IF DIVISIONAL AR DATA NOT IN’
000004
000005 EXECUTE ‘CHECK’
000006 ON_DATA
000007    IF ‘%%SYSTIME >= 100000 AND %%SYSTIE < 130000’
000008       DO PRINT DATA ‘DIV. INVOICES NOT IN, PLEASE CHECK IMMEDIATELY’
000009        DO PRINT DATA ‘********** ACTION REQUIRED WITHIN 30 MINUTES *******
   .
   .
   .
000029    ELSE
000030       DO PRINT DATA ‘TIME IS NOT BETWEEN 12 P.M. AND 6 P.M.’
000031       DO PRINT DATA ‘YOU’’RE OK FOR NOW, BUT DO NOT RELAX’
000032       DO SHOUT MSG ‘TIME IS NOT BETWEEN 12 P.M. AND 6 P.M. --- RELAX ****
000033 ********’ URGENCY ‘R’ DEST ‘TSO-M66’
000034       DO TERMINATE RESULT OK USER_CODE 0
```

The SYSRULE report is a formatted printout of the rule as it existed in the library member at the time it was executed. For example, it shows:

- the DO PRINT command as “DO PRINT DATA”
- the DO TERMINATE command as “DO TERMINATE RESULT”

These extra words, such as DATA and RESULT, help make the documentation easier to read.

Conclusion

You have now completed Problem 1 by defining and running your first rule. INTRAC is now better prepared, with its management being automatically warned when the divisional data arrives late.

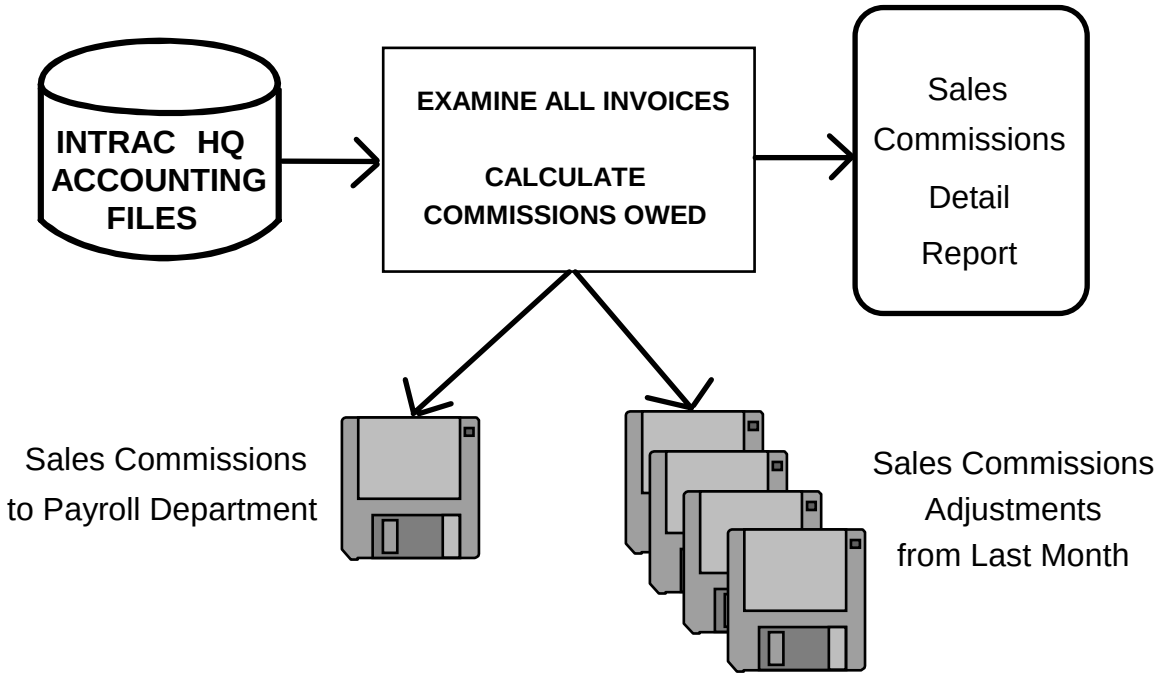
But your work has just begun. What you need to do next is check the commission report being sent to the payroll department and make sure it is OK.

Problem 2: Check Payroll Totals

The sales operations department makes up the annual sales plan, which contains the rules for calculating salesperson commissions. The calculations are complex.

Every month, the sales operations department runs their own computer program which calculates the commissions due to each salesperson in the company. This program produces a Sales Commission Detail report for the use of sales management. However, the sales operations department only provides the payroll department with the final amount of commissions earned by each sales employee, based on the report that they have prepared.

Figure 58 Payroll and Sales Calculation Flow



Problems

Every month, the payroll department has noticed that there are many adjustments to the commissions for the previous month. The payroll department has also heard, through the personnel department, of many complaints from the salespeople. The salespeople and management no longer trust the amounts they receive because of a history of mistakes. The payroll department is concerned that the file they receive, which contains commission amounts to be paid, has not been properly audited.

Solution — Overview

As a first step, the payroll department has decided to audit the Sales Commission Detail report for accuracy on the calculated totals. While the balancing activity of the payroll department, in the future, will be much more extensive, they need to implement their first balancing rule within three hours. This is when the payroll information is finalized for payment to employees. Because of the limited time available to implement this rule, they have decided to perform a simple check. But they will soon discover, when they implement the rule, that a major quality control surprise is waiting for them.

Figure 59 Sample Page from the Sales Commission Detail Report

```
----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9---
INTRAC INC.            SALES OPERATIONS DEPARTMENT     DATE 10/10/01 TIME 23:45      PAGE  4

    S A L E S     C O M M I S S I O N S    D E T A I L           FROM 10/10/01   TO  10/10 01
|SALESPERSON    HARVEY J. JONES  (197) | INVOICE # | INVOICE DATE |    AMOUNT   | STATUS
|                                      | CJ39394   | 02/02/00     |    2040.00 | COLLECTION |
|                                      | CJ98443   | 03/03/00     |     245.00 |            |
|                                      | MJ24545   | 04/04/00     |   10344.40 |            |
|                                      | RJ10030   | 01/02/00     |    5329.39 |            |
|                                      | RJ12233   | 02/02/00     |    5000.00 |            |
|                                      | RJ30021   | 03/02/00     |   16670.10 |            |
|                                      | RJ32991   | 02/03/00     |   30300.44 |            |
|                                      | RJ33939   | 03/03/00     |     900.25 |            |
|                                      | RJ38001   | 04/04/00     |    1000.23 |            |
|                                      | RJ93043   | 04/04/00     |     415.89 |            |
+-----------------------------------------------------------------------------------------
|TOTAL SALES - THIS MONTH                                                  72245.70 (PAID IN FULL)
|ADJUSTMENTS                                                               -5800.70
|YTD SALES (INCLUDING THIS MONTH)                                        -127879.00
|YTD COMMISSIONS PAID                                                           0.00
|YTD COMMISSIONS OWED                                                      -5115.00
|COMMISSION PAYROLL AMOUNT                                                  5115.00
-----------------------------------------------------------------------------------------
```

Shown above is a sample page from the Sales Commission Detail report. The audit of this report takes YTD Commissions paid, adds this to the current month commissions and compares the total to YTD Commissions owed. If the amounts are not equal, there is a serious payroll error.

To convince the INTRAC sales management of the integrity of the data, you would like to have a balancing report that shows the above calculations performed for every salesperson in the company.

You have obtained a small part of the production report, which is available in the IOA SAMPREPS library under the name CTBREP3. The part of the report on which you will work includes three sales employees and their actual figures from the last commission report.

Solution – Details

The rule consists of the following four ON blocks:

- INIT – initializes values of variables which you will use in the rule
- INPUT – extracts the sales data from the report
- CHECK – checks the data and prints a balancing report
- ENDJOB – prints a final comment and terminates the job

Opening the Rule

1 From the IOA Primary Option Menu, enter the Rule Definition facility by typing **BR** in the COMMAND field and pressing **Enter**. The Rule Definition entry panel is displayed:

Figure 60 CONTROL-M/Analyzer Rule Definition Entry Panel

```
-------------- CONTROL-M/Anlayzer RULE DEFINITION - ENTRY PANEL ------------(BR)
COMMAND ===>



SPECIFY LIBRARY, RULE

   LIBRARY ===> CTB.PROD.RULES
   RULE    ===>                  (Blank for rule selection list)



USE THE COMMAND "SHPF" TO SEE PFK ASSIGNMENT                          13.41.48
```

2 If the library name in the LIBRARY field in this example is different from your CONTROL-M / Analyzer RULES library name, change it to appear as shown in this example.

3 Leave the RULE field blank. Press **Enter**. The Rule List screen is displayed.

4 From the list of rules defined, select ARWARN02 for browsing as follows:

A Place the cursor in the OPT field, to the left of the rule named ARWARN02.

B Type **B** and press **Enter**. The Rule Definition screen for ARWARN02 is displayed.

Figure 61 Rule Definition Screen for Rule ARWARN02

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN02
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| OWNER   M66                      GROUP INTRAC                               |
| UPDATED 03/04/99  -  15:30:05    BY M66                                     |
| DESC    SUMMARIZE AND CHECK PAYROLL TOTALS                                  |
| OPTIONS                                                                     |
| =========================================================================== |
| EXECUTE INIT     UPON                                                     C  |
| ON DATA                                                                     |
| ALWAYS                                                                      |
|   DO SET      = ERRORCOUNT=0                                             C  |
|   DO SET      = SLSNO=0                                                  C  |
|   DO SET      = BLANK=' '                                                C  |
|   DO PRINT    = ******** PAYROLL CONTROL TOTALS *********           F   C  |
| =========================================================================== |
| EXECUTE INPUT    UPON                                                     C  |
| ON SYSOUT     PROCST          PGMST STEP01   DDNAME SYSUT2   JOBNM          |
|    MODE PG    LINECT 0000    DATASTAMP                                      |
| WHEN LINE 001     - 060     COL 001     - 025              STOP    AND/OR  |
|      STRING = 'YTD COMMISSIONS PAID'                                        |
|   DO EXTRACT  = YTDPAID                                                     |
|                 LEVEL 0 LINE +000 COL 0068 - 0078 PROCESS          TYP      |
PLEASE FILL IN RULE DEFINITION.                                       14.46.10
```

Defining ON Blocks for this Rule

5 The first ON block in this rule is called INIT, as defined in the line that begins “EXECUTE INIT.” This block is defined as an ON DATA block, set to ALWAYS perform the DO actions that follow it in the ON DATA block definition.

The purpose of this ON block is to initialize the values of variables that are used later in the rule. These variables are

- ERRORCOUNT, which is a variable to keep track of all of the errors encountered as the rule performs its balancing actions

  This count will be used, after examining the entire report, to determine if the rule should terminate OK or NOTOK.

- SLSNO, which is a sequential salesperson number

Because the report you are producing is distributed to various employees, and because the individual salesperson earnings printed on the report are confidential, the rule uses a salesperson number rather than a name. This number will be a sequential number assigned by you.

- BLANK, which is a variable that allows the rule to print blank lines, to create a better-looking report.

  Alternatively, you may use an AutoEdit variable named %%BLANK in your print line.

Many rules have ON blocks such as this one to initialize variables and perform other setup functions.

You can now look at the next ON block, called INPUT. To get a better look at this entire ON block, do the following to move its first line to the top of the screen:

6 Examine the SCROLL field in the upper right area of the screen. If this field does not contain the value CRSR, place the cursor in the SCROLL field and type **CRSR**.

7 To bring the first line of the INPUT block to the top of the screen, place the cursor on the line beginning with the words EXECUTE INPUT and press **PF08/PF20**, or enter the DOWN command. The first line of the block is now displayed at the top of the screen, and its entire contents are now visible.

Figure 62 INPUT Block

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN02
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| EXECUTE INPUT    UPON                                                    C   |
| ON SYSOUT     PROCST          PGMST STEP01   DDNAME SYSUT2   JOBNM          |
|    MODE PG    LINECT 0000    DATASTAMP                                      |
| WHEN LINE 001     - 060     COL 001     - 025              STOP    AND/OR   |
|      STRING = 'YTD COMMISSIONS PAID'                                        |
|   DO EXTRACT  = YTDPAID                                                     |
|                 LEVEL 0 LINE +000 COL 0068 - 0078 PROCESS          TYP      |
|   DO EXTRACT  = YTDOWED                                                     |
|                 LEVEL 0 LINE +001 COL 0068 - 0078 PROCESS          TYP      |
|   DO EXTRACT  = CURRENT                                                     |
|                 LEVEL 0 LINE +002 COL 0068 - 0078 PROCESS          TYP      |
|   DO BLOCK    = CHECK    ARG                                            C   |
| ========================================================================== |
| EXECUTE ENDJOB   UPON                                                   C   |
| ON DATA                                                                     |
| IF       ERRORCOUNT>0                                                   C   |
|   DO PRINT    = %%BLANK                                              F   C   |
|   DO PRINT    = ****** ERRORS IN COMMISSIONS ABOVE ******            F   C   |
|   DO TERMINAT = NOTOK    COD 0999                                           |
| ELSE                                                                        |
PLEASE FILL IN RULE DEFINITION.                                        15.00.48
```

Before examining this screen in detail, you need to first understand how a “data source” related ON block is structured.

Figure 63 ON Block Structure

```
ON BLOCK STRUCTURE

Data source related blocks:

EXECUTE <block name> UPON <execution criteria>
ON <data source>
WHEN    <criteria>
    <DO action>
    <DO action>
WHEN    <criteria>
    <DO action>
    <DO action>
```

When you need to extract data from a report or file, you use an ON block structure as shown in Figure 63. An ON block consists of the following parameters:

- EXECUTE – The EXECUTE parameter identifies the beginning of a new ON block. After the word EXECUTE, give the ON block a name. In these examples, you are not using the UPON parameter. This parameter can be used to specify criteria which determine if the current ON block is executed.

- ON <*data source*> – The ON parameter identifies the report or file that you want to process. What you specify here enables CONTROL-M / Analyzer to receive the output from the system.

- WHEN <*criteria*> – The WHEN parameter searches for the specific data that you need from your report.

- DO <*actions*> – DO parameters indicate the balancing actions to take for each line that met your search criteria.

8 Now see how this ON block structure is implemented in the INPUT ON block.

- EXECUTE INPUT marks the beginning of the block named INPUT.

- ON SYSOUT indicates that this block is working with a report file.

If you are not familiar with Job Control Language (JCL), the details for this part of the CONTROL-M / Analyzer rule can be provided by your system administrator. The ON SYSOUT parameter instructs CONTROL-M / Analyzer to look in the JCL in the first job step (STEP01). There, CONTROL-M / Analyzer will find a data definition (DD) JCL statement labeled SYSUT2. This is the statement that gives CONTROL-M / Analyzer the information it needs, in order to describe the characteristics of the report file.

- MODE PG indicates that the block is examining the report by page. This means that when data is extracted from the report, the line and column numbers are specified relative to the beginning of the current page.

- LINECT indicates the number of lines per page. When not specified, as above, it defaults to 60.

- WHEN describes the search criteria.

  As defined in this block, when the rule finds the data string "YTD COMMISSIONS PAID" anywhere between columns 1 through 25, in lines 1 through 60, meaning, on any line of the report, CONTROL-M / Analyzer is instructed to DO the balancing actions. The following DO EXTRACT and DO BLOCK instructions are performed repetitively, every time the WHEN search criteria are satisfied:

  — DO EXTRACT takes data out of the report and puts it into a variable. CONTROL-M / Analyzer can then reference and work with this variable. This extraction of data from the report occurs for each line in which the string "YTD COMMISSIONS PAID" is found.

Figure 64 Extract from the Sales Commission Detail Report

```
----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9---
INTRAC INC.             SALES OPERATIONS DEPARTMENT     DATE 10/10/01 TIME 23:45      PAGE  4

    S A L E S     C O M M I S S I O N S    D E T A I L          FROM 10/10/01   TO  10/10 01
|SALESPERSON    HARVEY J. JONES  (197)   | INVOICE #  | INVOICE DATE |  AMOUNT   | STATUS
|                                        | CJ39394    | 02/02/00     |   2040.00| COLLECTION|
|                                        | CJ98443    | 03/03/00     |    245.00|           |
|                                        | MJ24545    | 04/04/00     |  10344.40|           |
|                                        | RJ10030    | 01/02/00     |   5329.39|           |
|                                        | RJ12233    | 02/02/00     |  5000.00 |           |
|                                        | RJ30021    | 03/02/00     | 16670.10 |           |
|                                        | RJ32991    | 02/03/00     | 30300.44 |           |
|                                        | RJ33939    | 03/03/00     |   900.25 |           |
|                                        | RJ38001    | 04/04/00     |  1000.23 |           |
|                                        | RJ93043    | 04/04/00     |   415.89 |           |
+-----------------------------------------------------------------------------------------------
|TOTAL SALES - THIS MONTH                                                  72245.70 (PAID IN FULL
|ADJUSTMENTS                                                               -5800.00
|YTD SALES (INCLUDING THIS MONTH)                                         -127879.00
|YTD COMMISSIONS PAID                                                           0.00
|YTD COMMISSIONS OWED                                                      -5115.00
|COMMISSION PAYROLL AMOUNT                                                  5115.00
-----------------------------------------------------------------------------------------------
```

— DO EXTRACT YTDPAID—The rule extracts report data from columns 68-78 into a variable called YTDPAID. The report data comes from the above WHEN search, specifically line +0, meaning the line in which “YTD COMMISSIONS PAID” is found, plus 0 lines.

— DO EXTRACT YTDOWED—This information is also extracted from columns 68-78 into the variable called YTDOWED. The report data comes from line +1, meaning the line in which “YTD COMMISSIONS PAID” is found, + 1 line.

— DO EXTRACT CURRENT—This information also comes from the report data columns 68-78 and is put into the variable called YTDOWED. The report data comes from line +2, meaning the line in which “YTD COMMISSIONS PAID” is found, + 2 lines.

— DO BLOCK CHECK — After extracting the above information, perform ON block CHECK before continuing to search further down the report for the next occurrence of “YTD COMMISSIONS PAID.”

Now that the information has been extracted, the rule performs balancing actions on the data. Therefore, a separate ON DATA block is defined in to perform the balancing actions.

9 Using **PF08/PF20**, or entering the DOWN command, scroll forward until you see the ON block called EXECUTE CHECK.

10 Place the cursor on the line beginning with EXECUTE CHECK and press **PF08/PF20** or enter the DOWN command.

Figure 65 The EXECUTE CHECK ON Block

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN02
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| ========================================================================== |
| EXECUTE CHECK    UPON                                                    C   |
| ON DATA                                                                  |
| ALWAYS                                                                   |
|   DO SET      = CALCAMT=YTDPAID+CURRENT                                  C   |
|   DO SET      = DIFFERENCE=ABS(CALCAMT-YTDOWED)                          C   |
|   DO SET      = SLSNO=INT(SLSNO+1)                                       C   |
|   DO PRINT    = SALESPERSON #%%SLSNO                                 F   C   |
|   DO PRINT    = YTD COMM. PAID             %%YTDPAID                 F   C   |
|   DO PRINT    = CURRENT AMOUNT             %%CURRENT                 F   C   |
|   DO PRINT    = ----------------------------------------             F   C   |
|   DO PRINT    = CALCULATED TOTAL              %%CALCAMT              F   C   |
|   DO PRINT    = REPORTED TOTAL             %%YTDOWED                 F   C   |
|   DO PRINT    = ----------------------------------------             F   C   |
|   DO PRINT    = DIFFERENCE   ***** %%DIFFERENCE *****                F   C   |
|   DO PRINT    = %%BLANK                                              F   C   |
| IF       DIFFERENCE>0                                                    C   |
|   DO PRINT    = ***** SERIOUS PAYROLL ERROR - SEE ABOVE *****        F   C   |
|   DO PRINT    = %%BLANK                                              F   C   |
|   DO SET      = ERRORCOUNT=ERRORCOUNT+1                                  C   |
PLEASE FILL IN RULE DEFINITION.                                      15.06.39
```

The ON block called CHECK performs calculations and prints information to the balancing report. Generally, calculations are performed using DO SET parameters. DO SET assigns a value to a variable. Output to the report is performed with DO PRINT parameters.

The ON block CHECK performs the following actions:

- The new variable CALCAMT is set to the sum of YTDPAID and CURRENT. These two variables were extracted from the report.

- The variable DIFFERENCE is calculated as the difference between the amount just calculated (CALCAMT) and the amount from the report (YTDOWED). As you will see later in the rule, if there is a difference, it means that there is a serious error. Function ABS is used to set the difference to the “absolute value” of CALCAMT minus YTDOWED.

  The ABS function is one of the many functions available in CONTROL-M / Analyzer to assist the balancing effort. The ABS function is used because the difference should always print as a positive number. When there is a difference, you do not know which amount is the correct amount (CALCAMT or YTDOWED). All you are concerned with in this instance is that there is a difference.

- The last DO SET parameter increments the salesperson number (SLSNO) by 1. If you want to examine the commission totals for more than one salesperson, you can continue sequentially through the current list of salespeople, based on their salesperson numbers.

- The report is printed with the group of DO PRINT parameters. As you may recall from the first rule as defined in Problem 1, these print lines are output to two places: the Invocation Report on SYSPRINT, and the user report on SYSUSER.

- Whenever you want to include a variable in the print line, precede the variable name with two percent signs (%%). As you see in Figure 65, this ON block prints several lines and several variables: YDTPAID, CURRENT, CALCAMT, YTDOWED, DIFFERENCE.

You can compare these instructions to the actual SYSUSER report, as shown in Figure 68.

The last part of this ON block contains an IF and ELSE combination of parameters.

11 Position the cursor on the line beginning with the string “IF DIFFERENCE > 0” and press **PF08/PF20**, or enter the DOWN command.

Figure 66 IF and ELSE Parameters

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN02
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| IF       DIFFERENCE>0                                                   C   |
|   DO PRINT    = ***** SERIOUS PAYROLL ERROR - SEE ABOVE *****       F   C   |
|   DO PRINT    = %%BLANK                                             F   C   |
|   DO SET      = ERRORCOUNT=ERRORCOUNT+1                                 C   |
| ELSE                                                                      |
|   DO PRINT    = COMMISSIONS OK                                      F   C   |
|   DO PRINT    = %%BLANK                                             F   C   |
| ========================================================================== |
| EXECUTE          UPON                                                     C   |
| ON                                                                        |
======= >>>>>>>>>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<<<<<< =====









PLEASE FILL IN RULE DEFINITION.                                        15.07.59
```

This IF and ELSE definition determines for each salesperson if there is a problem in the balancing report, as follows:

- If the DIFFERENCE that was calculated previously was greater than 0, then

  — print an error message, followed by a blank line

  — add 1 to the error count

  otherwise (ELSE),

- Print a message indicating that the commissions are OK, followed by a blank line

Terminating the Rule

12 All that is left is for the rule definition to terminate it after the entire report has been examined. This is done with the ON block called ENDJOB.

A Press **PF07/PF19** or enter the UP command, to scroll backward until you see the ON block ENDJOB.

B Place the cursor on the line beginning with the string “EXECUTE ENDJOB” and press **PF08/PF20** or enter the DOWN command.

Figure 67 ON Block ENDJOB

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN02
COMMAND ===>                                                     SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| EXECUTE ENDJOB   UPON                                                    C   |
| ON DATA                                                                      |
| IF       ERRORCOUNT>0                                                    C   |
|   DO PRINT    = %%BLANK                                              F   C   |
|   DO PRINT    = ****** ERRORS IN COMMISSIONS ABOVE ******            F   C   |
|   DO TERMINAT = NOTOK    COD 0999                                            |
| ELSE                                                                         |
|   DO PRINT    = %%BLANK                                              F   C   |
|   DO PRINT    = ---------  COMMISSIONS ARE PERFECT  ---------        F   C   |
|   DO TERMINAT = OK       COD 0000                                            |
| ========================================================================== |
| EXECUTE CHECK    UPON                                                    C   |
| ON DATA                                                                      |
| ALWAYS                                                                       |
|   DO SET      = CALCAMT=YTDPAID+CURRENT                                  C   |
|   DO SET      = DIFFERENCE=ABS(CALCAMT-YTDOWED)                          C   |
|   DO SET      = SLSNO=INT(SLSNO+1)                                       C   |
|   DO PRINT    = SALESPERSON #%%SLSNO                                 F   C   |
|   DO PRINT    = YTD COMM. PAID            %%YTDPAID                  F   C   |
|   DO PRINT    = CURRENT AMOUNT            %%CURRENT                  F   C   |
PLEASE FILL IN RULE DEFINITION.                                        15.08.50
```

Because ON block ENDJOB is the block that sequentially follows ON block INPUT, it is executed right after CONTROL-M / Analyzer has finished executing ON block INPUT—that is, after the rule has scanned the entire report and found and processed all of the lines it was searching for.

ON block ENDJOB performs one of the following two actions, depending on whether there were any errors found when the data was checked:

- If there any errors were found, the rule prints a final line to indicate errors, and then terminates the rule with a NOTOK result and with the user code of 999. This code is a numeric code that indicates an out-of-balance situation.

NOTE

You can create your own rules, and you can choose your own codes to signify various errors.

- If no errors are found, this indicates that the commissions are perfect in the printout. The rule terminates with an OK result, and with the user code of 0.

Conclusion

You have now completed Problem 2 by examining the definition of the second rule. In the IOA SAMPLE library there is a member named RUNRULE2 that runs the rule you just examined. You may submit this job and review the output.

Figure 68 Output from Job RUNRULE2

```
******** PAYROLL CONTROL TOTALS ********
SALESPERSON #1
YTD COMM. PAID                    0.00
CURRENT AMOUNT                 5115.00
----------------------------------------
CALCULATED TOTAL               5115.000
REPORTED TOTAL                -5115.00
----------------------------------------
DIFFERENCE   ***** 10230.00 *****

***** SERIOUS PAYROLL ERROR - SEE ABOVE
*****

SALESPERSON #2
YTD COMM. PAID                 1883.13
CURRENT AMOUNT                11632.03
----------------------------------------
CALCULATED TOTAL              13515.160
REPORTED TOTAL                13515.16
----------------------------------------
DIFFERENCE   ***** 0.0000 *****

COMMISSIONS OK

SALESPERSON #3
YTD COMM. PAID                    0.00
CURRENT AMOUNT                39995.16
----------------------------------------
CALCULATED TOTAL              39995.160
REPORTED TOTAL                39995.16
----------------------------------------
DIFFERENCE   ***** 0.0000 *****

COMMISSIONS OK
```

Looking at the results for Salesperson #1, you can see that the balancing rule has caught a significant error. This individual earned negative commissions. Perhaps this is a new salesperson who has had no sales, but only returns. For whatever reason, this salesperson's commissions were -$5,115. The amount specified to the payroll department should probably have been 0. Instead, the amount specified was $5,115! Fortunately, you caught this mistake in time.

Problem 3: Check Sales Commissions

The payroll user has been thinking about the congratulations that he received from the managing director of INTRAC. Because of the quick implementation of rule ARWARN02 from Problem 2, he has saved the company time and money. With this quick success, the payroll user moves ahead to work on to Problem 3.

There is one more simple check that the payroll department will perform on the Sales Commissions Detail report. In this rule, a new feature of CONTROL-M / Analyzer is introduced — the ability to summarize extracted information.

Look again at a sample page of the report.

Figure 69 Sales Commissions Detail Report

```
----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9---
INTRAC INC.             SALES OPERATIONS DEPARTMENT     DATE 10/10/01 TIME 23:45      PAGE  4

    S A L E S     C O M M I S S I O N S    D E T A I L           FROM 10/10/01   TO  10/10 01
| SALESPERSON    HARVEY J. JONES  (197) | INVOICE # | INVOICE DATE |    AMOUNT   | STATUS
|                                       | CJ39394   | 02/02/00     |    2040.00 | COLLECTION|
|                                       | CJ98443   | 03/03/00     |     245.00 |           |
|                                       | MJ24545   | 04/04/00     |   10344.40 |           |
|                                       | RJ10030   | 01/02/00     |    5329.39 |           |
|                                       | RJ12233   | 02/02/00     |    5000.00 |           |
|                                       | RJ30021   | 03/02/00     |   16670.10 |           |
|                                       | RJ32991   | 02/03/00     |   30300.44 |           |
|                                       | RJ33939   | 03/03/00     |     900.25 |           |
|                                       | RJ38001   | 04/04/00     |    1000.23 |           |
|                                       | RJ93043   | 04/04/00     |     415.89 |           |
+-----------------------------------------------------------------------------------------------
|TOTAL SALES - THIS MONTH                                                 72245.70 (PAID IN FULL)
|ADJUSTMENTS                                                              -5800.70
|YTD SALES (INCLUDING THIS MONTH)                                        -127879.00
|YTD COMMISSIONS PAID                                                          0.00
|YTD COMMISSIONS OWED                                                     -5115.00
|COMMISSION PAYROLL AMOUNT                                                 5115.00
-----------------------------------------------------------------------------------------------
```

Problem

Based on the error that the payroll department found from using balancing Rule 2 (ARWARN02), the payroll department users are concerned about the basic integrity of the Sales Commission Detail report. They need to develop some control totals on the report and compare these totals to each other and to published accounting department figures.

Solution - Overview

The users will check that the grand total of all of the invoice details for all salespeople adds up to the grand total of Total Sales for all salespeople. They can compare these figures to figures that the accounting department has officially published.

Solution - Details

The rule consists of the following three ON blocks:

- INIT – initializes values of variables in the rule
- INPUT – extracts and summarizes the above data from the report
- FINISH – checks the final totals and prints a balancing report

1 From the Rule List screen, select rule ARWARN03 for browsing.

If you are not familiar with how to select a rule for browsing, review the instructions at the beginning of the discussion of rule ARWARN02 in Problem 2.

Figure 70 Rule Definition Screen for Rule ARWARN03

```
        LIBRARY : CTB.PROD.RULES                              RULE : ARWARN03
COMMAND ===>                                                   SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| OWNER   M66                      GROUP INTRAC                               |
| UPDATED 03/04/99  -  15:30:05    BY M66                                     |
| DESC    SUMMARIZE AND CHECK SALES COMMISSIONS                               |
| OPTIONS                                                                     |
| =========================================================================== |
| EXECUTE INIT     UPON                                                     C  |
| ON DATA                                                                     |
| ALWAYS                                                                      |
|   DO SET      = ERRORCOUNT=0                                             C  |
|   DO SET      = BLANK=' '                                                C  |
| =========================================================================== |
| EXECUTE INPUT    UPON                                                     C  |
| ON SYSOUT     PROCST          PGMST STEP01   DDNAME SYSUT2   JOBNM          |
|    MODE PG    LINECT 0000    DATASTAMP                                      |
| WHEN LINE 001     - 060     COL 050     - 060              STOP    AND/OR  |
|      STRING = /                                                             |
|   DO EXTRACT  = SUM1                                                        |
|                 LEVEL 0 LINE +000 COL 0068 - 0078 PROCESS          TYP SU  |
| WHEN LINE 001     - 060     COL 001     - 020              STOP    AND/OR  |
|      STRING = TOTAL SALES                                                   |
PLEASE FILL IN RULE DEFINITION.                                        15.14.40
```

2 Place the cursor on the line beginning with the string “EXECUTE INPUT,” and press **PF08/PF20**, or execute the DOWN command. The first line of the ON block INPUT is now displayed at the top of the screen.

Figure 71 ON Block INPUT

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN03
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| EXECUTE INPUT    UPON                                                     C   |
| ON SYSOUT     PROCST          PGMST STEP01   DDNAME SYSUT2   JOBNM          |
|    MODE PG    LINECT 0000    DATASTAMP                                      |
| WHEN LINE 001     - 060     COL 050     - 060              STOP    AND/OR   |
|      STRING = /                                                             |
|   DO EXTRACT  = SUM1                                                        |
|                 LEVEL 0 LINE +000 COL 0068 - 0078 PROCESS          TYP SU   |
| WHEN LINE 001     - 060     COL 001     - 020              STOP    AND/OR   |
|      STRING = TOTAL SALES                                                   |
|   DO EXTRACT  = SUM2                                                        |
|                 LEVEL 0 LINE +000 COL 0068 - 0078 PROCESS          TYP SU   |
| ========================================================================== |
| EXECUTE FINISH   UPON                                                    C   |
| ON DATA                                                                     |
| ALWAYS                                                                      |
|   DO PRINT    = ****** COMMISSION CONTROL TOTALS *******            F   C   |
|   DO PRINT    = REPORTED SALES                 %%SUM2               F   C   |
|   DO PRINT    = CALCULATED SALES               %%SUM1               F   C   |
|   DO PRINT    = ----------------------------------------            F   C   |
|   DO SET      = DIFFERENCE1=SUM2-SUM1                                  C   |
PLEASE FILL IN RULE DEFINITION.                                      15.17.24
```

The ON block INPUT performs the following actions:

- The ON SYSOUT definition is identical to the one in rule ARWARN02. It provides the connection to the report file stored on the system.

- The first WHEN definition selects all report lines that have a “ / ” between columns 50 and 60. The “ / ” is part of a date field. This is how detail lines are identified from the report (see Figure 69). From these detail lines, the rule extracts the information in columns 68 through 78 and puts it into a variable called SUM1.

- The second WHEN definition selects the report lines that contain the string TOTAL SALES in columns 1 through 20. This is the line containing the total sales amount of each sales employee. From these report lines, the rule extracts the information in columns 68 through 78 and puts it into a variable called SUM2.

  Parameter TYP for this WHEN definition is set to SU. This indicates that the rule should accumulate a sum or average of the information being extracted. In the rule, the sum of each of the report line amounts is to be placed in their respective variables, SUM1 and SUM2. This means that after the entire report is processed, SUM1 will contain the grand total of invoice amounts from all report lines with a “ / ” in columns 50 through 60. SUM2 will contain the grand total of all report line amounts from the TOTAL SALES lines.

After the report file is processed, the next sequential ON block, called FINISH, is executed.

3 Place the cursor on the line beginning with the string, EXECUTE FINISH, and press **PF08/PF20**, or enter the DOWN command.

Figure 72 ON Block EXECUTE FINISH

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN03
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| EXECUTE FINISH   UPON                                                   C   |
| ON DATA                                                                     |
| ALWAYS                                                                      |
|   DO PRINT    = ****** COMMISSION CONTROL TOTALS *******            F   C   |
|   DO PRINT    = REPORTED SALES                %%SUM2                F   C   |
|   DO PRINT    = CALCULATED SALES              %%SUM1                F   C   |
|   DO PRINT    = ----------------------------------------           F   C   |
|   DO SET      = DIFFERENCE1=SUM2-SUM1                                 C   |
|   DO PRINT    = DIFFERENCE                        %%DIFFERENCE1     F   C   |
|   DO PRINT    = %%BLANK                                             F   C   |
|   DO PRINT    = %%BLANK                                             F   C   |
| IF       DIFFERENCE1>1000                                               C   |
|   DO PRINT    = ***** SERIOUS COMMISSION ERROR - SEE ABOVE *****    F   C   |
|   DO TERMINAT = NOTOK    COD 0999                                           |
| ELSE                                                                        |
|   DO PRINT    = COMMISSIONS OK                                      F   C   |
|   DO TERMINAT = OK       COD 0000                                           |
| ========================================================================== |
| EXECUTE          UPON                                                   C   |
| ON                                                                          |
PLEASE FILL IN RULE DEFINITION.                                      15.18.49
```

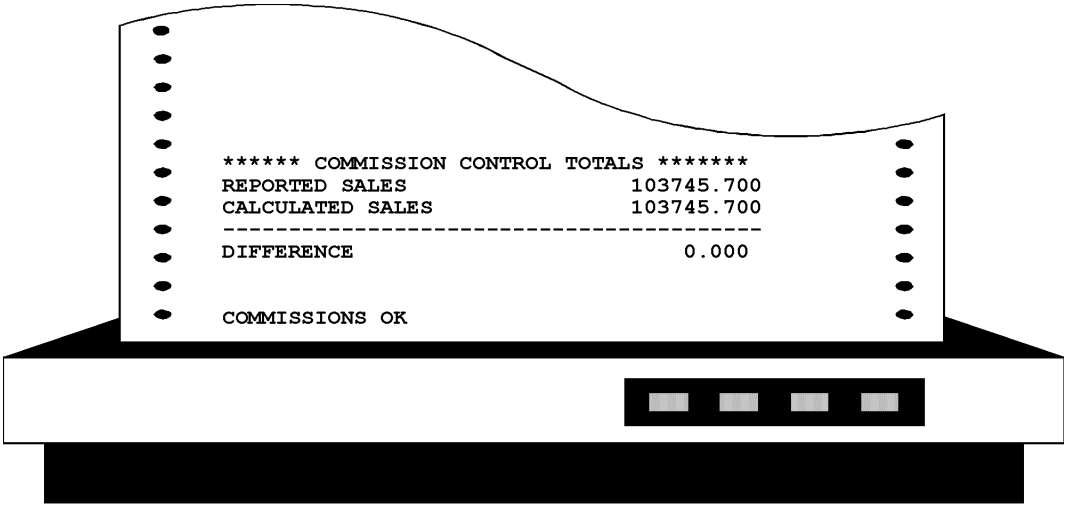
The ON block FINISH performs the following actions:

- Prints a report heading and the two sums, SUM1 and SUM2, that were calculated with the TYP SU parameter.
- Calculates the difference between the two variables and then prints the difference, followed by two blank lines.
- To end the rule, determine, with the IF parameter, if the difference is tolerable or not. If the difference is greater than $1000, call this a serious commission error and terminate the rule NOTOK, with a 999 user code. Otherwise, indicate that these totals are OK and terminate the rule accordingly, terminating with OK, and with a user code of 0.

This concludes the definition for Rule ARAWARN03.

Look at the SYSUSER report that this rule produced. You may use the job RUNRULE3 provided in the CTB JCL library to run this rule. You may then review the results online, using SDSF, ROSCOE, or a similar facility.

Figure 73 SYSUSER Report from RUNRULE3



The rule has examined the entire report, line by line, accumulated totals, and found that the details match 100% with the reported totals.

Conclusion

You have now examined and implemented three basic rules. The parameters and ON blocks that you have just learned about form the basis for continuing with the more advanced CONTROL-M / Analyzer features. These advanced features are discussed in the next section, “Advanced Problems and Solutions” on page 2-108

If your purpose in reviewing this guide was to gain an overview of CONTROL-M / Analyzer capabilities, you may want to finish here.

If you would like to explore more advanced examples of CONTROL-M / Analyzer rules, including

- processing of sequential files
- use of database generations
- use of more sophisticated input data screening
- use of other functions

then, proceed to the next page.

Advanced Problems and Solutions

Problem 4: Balance Divisional Accounting Files

Problem

Without an automated data balancing product, your accounting user is spending hours every day looking through large reports for errors. It is a nearly impossible task. INTRAC generates hundreds of invoices daily. The user had to scan through the lists of invoices, often under the pressures of other tasks, and limited time. Most of the errors are not obvious, but each error that creeps into the system is costly to correct.

Solution — Overview

Based on an analysis of the types of errors encountered over the past year, the user has set criteria for accepting the daily invoice files from the divisions. The criteria are shown in the file description. See “Sources of Data—Reports and Files” on page 2-72.

1. The daily invoice file must be verified, before being passed on to the accounting department for processing. The verification will include all the items specified in the file description. These include items such as invoice date OK, division # OK, amounts OK, and totals are in balance with the details.

2. When the file is verified, the two totals (Invoice Amount and Cost) must be saved, for each of the three divisions—meaning that six variables are calculated. These amounts will be used for weekly reasonableness checks, which are defined in rule ARWARN05.

3. Any data not meeting the balancing criteria must be reported.

Solution - Details

In this CONTROL-M / Analyzer solution, the rule

- analyzes the divisional invoice file, checking each row or record of data
- reports errors
- accumulates divisional totals for invoice amounts and cost
- updates the CONTROL-M / Analyzer database if all of the errors are “tolerable,” or if there are no errors

In this rule, you will learn several new features, including

- using a sequential file as a data source
- committing database variables
- using the SUBSTR and INT functions
- using the %% prefix in defining variable names

Opening Rule ARWARN04

1 Select rule ARWARN04 for browsing.

If you are not familiar with how to select a rule for browsing, review the instructions at the beginning of the discussion of rule ARWARN02 in Problem 2.

Figure 74 Rule ARWARN04

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN04
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| OWNER   M66                       GROUP INTRAC                               |
| UPDATED 20/12/99  -  15:02:19     BY M66                                     |
| DESC    ANALYZE AND REPORT ON DIV ARDATA FILES                               |
| OPTIONS                                                                      |
| ============================================================================ |
| EXECUTE INIT     UPON                                                      C  |
| ON DATA                                                                      |
| ALWAYS                                                                       |
|   DO PRINT    =         INTRAC DAILY DIVISIONAL INVOICE ANALYSIS    F   C   |
|   DO PRINT    =                     FOR  %%SYSDATE                  F   C   |
|   DO PRINT    = -- ------- -------- ------- ------- --------- ----- F   C Y |
|             ----                                                             |
|   DO PRINT    = DV CUST    CUST #   INV #   DATE    AMOUNT    PROBL F   C Y |
|             EM                                                               |
|   DO PRINT    = -- ------- -------- ------- ------- --------- ----- F   C Y |
|             ----                                                             |
|   DO SET      = ERRORS=0                                                 C   |
|   DO SET      = TOLERABLES=0                                             C   |
|   DO SET      = BLANK=' '                                                C   |
| ============================================================================ |
PLEASE FILL IN RULE DEFINITION.                                        15.22.36
```

The rule consists of the following five ON blocks:

- INIT – Prints initial report headings and initializes values of variables in the rule
- INPUT – Extracts data from the sequential divisional invoice file, recognizing two types of records in the file—detail records and total records
- CHKDETL – Analyzes detail records for errors
- PRINT – Prints the detail balancing report

- ENDJOB – Analyzes the divisional totals, commits the Database variables, if there are no intolerable errors, and prints the rest of the balancing report

2 Place the cursor on the line beginning with the string “EXECUTE INIT,” and press **PF08/PF20**, or enter the DOWN command.

Figure 75 EXECUTE INIT Block

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN04
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| EXECUTE INIT     UPON                                                   C   |
| ON DATA                                                                     |
| ALWAYS                                                                      |
|   DO PRINT    =         INTRAC DAILY DIVISIONAL INVOICE ANALYSIS    F   C   |
|   DO PRINT    =                     FOR  %%SYSDATE                  F   C   |
|   DO PRINT    = -- ------- -------- ------- ------- --------- ----- F   C Y |
|             ----                                                            |
|   DO PRINT    = DV CUST    CUST #   INV #   DATE    AMOUNT    PROBL F   C Y |
|             EM                                                              |
|   DO PRINT    = -- ------- -------- ------- ------- --------- ----- F   C Y |
|             ----                                                            |
|   DO SET      = ERRORS=0                                                C   |
|   DO SET      = TOLERABLES=0                                            C   |
|   DO SET      = BLANK=' '                                               C   |
| ========================================================================== |
| EXECUTE INPUT    UPON                                                   C   |
| ON FILE       FILENAME IOAP.V610.SAMPREPS(CTBREP1)                          |
|    MODE DS    LINECT 0000    DATASTAMP                                      |
| WHEN LINE 001     - 999     COL 003     - 015             STOP    AND/OR   |
|      STRING = .NE.TOTALINVOICES                                             |
PLEASE FILL IN RULE DEFINITION.                                       15.24.47
```

The DO PRINT instructions print the report heading lines.

As in previous rules you examined, this rule defines a variable that accumulates the error count, in this case, ERRORS. Another variable, TOLERABLES, that stores the accumulated number of tolerable errors, is defined. Tolerable errors will result in warning messages appearing on the report, but will still result in an OK condition and with the database being updated.

3 Place the cursor on the line beginning with the string “EXECUTE INPUT,” and press **PF08/PF20**, or enter the DOWN command.

Figure 76 EXECUTE INPUT Block

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN04
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| EXECUTE INPUT    UPON                                                     C  |
| ON FILE       FILENAME IOAP.V610.SAMPREPS(CTBREP1)                          |
|    MODE DS    LINECT 0000    DATASTAMP                                      |
| WHEN LINE 001     - 999     COL 003     - 015              STOP    AND/OR  |
|      STRING = .NE.TOTALINVOICES                                             |
|   DO EXTRACT  = DIV                                                         |
|                 LEVEL 0 LINE +000 COL 0001 - 0002 PROCESS          TYP      |
|   DO EXTRACT  = CUST                                                        |
|                 LEVEL 0 LINE +000 COL 0003 - 0030 PROCESS          TYP      |
|   DO EXTRACT  = CUSTNO                                                      |
|                 LEVEL 0 LINE +000 COL 0031 - 0038 PROCESS          TYP      |
|   DO EXTRACT  = INVNO                                                       |
|                 LEVEL 0 LINE +000 COL 0039 - 0045 PROCESS          TYP      |
|   DO EXTRACT  = DATE                                                        |
|                 LEVEL 0 LINE +000 COL 0046 - 0051 PROCESS          TYP      |
|   DO EXTRACT  = AMT                                                         |
|                 LEVEL 0 LINE +000 COL 0052 - 0061 PROCESS          TYP      |
|   DO EXTRACT  = AMT%%DIV                                                    |
|                 LEVEL 0 LINE +000 COL 0052 - 0061 PROCESS          TYP SU   |
|   DO EXTRACT  = COST                                                        |
PLEASE FILL IN RULE DEFINITION.                                       15.25.36
```

The value in the INPUT ON block for a file has the identical structure to the INPUT ON block for a report. As a reminder, the structure is:

- ON – to specify the data source
- WHEN – to specify the search criteria
- DO – to specify balancing actions

ON Parameter

The ON parameter specifies the data source as a file and provides the file name. The rule can process a file in one of the following two MODEs:

- PG (page) – A rule can process a file as if it were a series of pages of a report. In such a case, the rule may optionally define a line count.

- DS (dataset) – If the file does not have the appearance of a report, meaning that it is not organized into pages, the rule can process a file as a series of records or rows of a dataset.

This rule processes the invoice file in DS mode.

WHEN Criteria

In this file, two WHEN criteria describe the two types of rows on which the rule needs to take action:

- Detail Rows, which do not contain the string, TOTALINVOICES, in columns 3 through 15.
- Total Rows, which contain the string, TOTALINVOICES, in columns 3 through 15.

DO Parameters

The DO EXTRACT parameters extract information from each Detail Row of the file when the search criteria specified in the WHEN parameter are satisfied. Shown below is a sample of the data in an extracted row (Figure 77), corresponding to each of the DO EXTRACT parameters that are specified in the rule:

Figure 77 Sample Data for DO EXTRACT Definition

```
Div                            Customer---------------- Invoice -----------------
# Customer Name                  #       #   -Date-  Amount    Cost       Misc.
01ABC COMPANY                  1ABC0001CJ39394020892   2040.00   1800.00   157.95
01SHANGHAI MOTORS              1SHA0002CJ39395030892    245.00     30.00    10.00
01TELEPATHIC ENTERPRISES       1TEL0001CJ39396040892  10987.55  10005.00   300.00
01CBG COMPANY                  1CBG0001CJ39397020892  12044.00  12000.00     0.00
01TERRIBLE TRUCKERS            1TER0001CJ39398030892 345245.00 308000.00  4750.00
01BIG DEAL COMPUTING LTD       1BIG0001CJ39399040892    987.25    600.00     0.00
01UNDERGROUND TELEPHONES INC   1UND0001CJ39400020892     40.00     20.00     0.00
01MITSUISUN ELECTRONICS        1MIT0001CJ39401030892   6776.45   6000.00   115.00
01TOTALINVOICES                                       378365.25 338455.00  6789.00
02SWAHILI ARMED FORCES         2SWA0001MJ24545040892  10987.55  10005.00   275.00
02GOVERNMENT OF BOSSNIA        2GBO0001MJ24546020892  27500.00  26800.00   200.50
02METZUYAN MANUFACTURING       2MET0001MJ24547030892  13465.00  12830.00   347.00
02TELEPATHIC ENTERPRISES       2TEL0002MJ24548040892   5590.00   5600.00     0.00
02BANGUP AIRCRAFT              2BAN0001MJ24549020892   2150.00   1800.00     7.00
02ZAPIT HYDRO & ELECTRIC       2ZAP0001MJ24550030892  45222.00  40900.00   815.00
02POST OF THE AMERICAS         2POS0001MJ24551040892  10000.50   8005.00   500.00
02GROSS AIRLINES               2GRO0001MJ24552020892    197.75    100.00     0.00
----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
```

Note the following details in the second to last DO parameter in Figure 76 (DO EXTRACT = AMT%%DIV):

- The variable being extracted (AMT%%DIV) contains the %% AutoEdit operator symbol, resulting in the following:

  — When the rule extracts data from the current line in the file and performs this DO EXTRACT procedure, the rule substitutes the current value of variable DIV (division) in place of the entire string %%DIV.

    For example, if the rule were looking at the first row in the report, the current value of the DIV variable would be set to “01” (see the first line in Figure 77).

  — The name of the variable extracted would then be set to AMT01, substituting 01 for %%DIV in the variable name AMT%%DIV.

  For more details about AutoEdit variables, see the *CONTROL-M/Analyzer for z/OS User Guide*.

- The AMT%%DIV variable is set to TYP SU. This means that the rule will accumulate a sum for each division. Specifically, after the entire file is processed, the resulting variables will be: AMT01, AMT02, and AMT03 (see Figure 77). Each variable will contain the sum of that division's invoice amounts. By scrolling forward through the report using **PF08/PF20** (**DOWN**), you will see that COST divisional totals are accumulated also.

4 Now examine the remainder of this INPUT block. Use **PF08/PF20**, or enter the DOWN command, and scroll forward to the DO SET parameter set to NAME=SUBSTR(1,7CUST), immediately following the DO EXTRACT parameters.

5 Place the cursor on the line beginning with DO SET = NAME, and press **Enter**.

Figure 78 DO SET Parameters

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN04
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
|   DO SET      = NAME=SUBSTR(1,7,CUST)                                   C   |
|   DO SET      = MONTHYR=SUBSTR(3,4,DATE)                                C   |
|   DO BLOCK    = CHKDETL  ARG                                            C   |
| WHEN LINE 001     - 999     COL 003     - 015              STOP    AND/OR   |
|      STRING = TOTALINVOICES                                                  |
|   DO EXTRACT  = DIV                                                          |
|                 LEVEL 0 LINE +000 COL 0001 - 0002 PROCESS          TYP      |
|   DO EXTRACT  = TAMT%%DIV                                                    |
|                 LEVEL 0 LINE +000 COL 0052 - 0061 PROCESS          TYP      |
|   DO EXTRACT  = TCOST%%DIV                                                   |
|                 LEVEL 0 LINE +000 COL 0062 - 0071 PROCESS          TYP      |
| ========================================================================== |
| EXECUTE ENDJOB   UPON                                                   C   |
| ON DATA                                                                     |
| ALWAYS                                                                      |
|   DO SET      = I=1                                                     C   |
| LABEL: TOTALS                                                               |
| IF       I GT 3                                                         C   |
|   DO GOTO     LABEL FINISH                                                  |
| ELSE                                                                        |
PLEASE FILL IN RULE DEFINITION.                                       15.28.55
```

These last parameters of the rule are defined as follows:

- The first DO SET instruction takes the first 7 characters of the customer name, CUST, using the SUBSTR function, and assigns them to a variable called NAME. This illustrates the use of the substring function in order to save space in the report line. The SUBSTR function syntax requires the starting position in a field, in this case “1,” the number of characters to extract, in this case “7,” and the field name, in this case “CUST.”

- The next DO SET instruction creates a variable called MONTHYR, comprised of positions 3 through 6 of the variable DATE, which the rule had extracted from the file. These positions represent the invoice month and year, which must be checked for validity.

- Having extracted all the information needed, the rule executes an ON block within the rule called CHKDETL. This ON block is explained in “Examining the CHKDETL Block” on page 2-115.

- The second and last WHEN parameter of this rule searches for the database row containing TOTALINVOICES in columns 3 through 15.

When the rule finds the desired rows, it extracts three variables using DO EXTRACT actions: DIV, TAMT%%DIV, and TCOST%%DIV.

When CONTROL-M / Analyzer reads the string %%DIV, it substitutes the value of the variable DIV in place of the string %%DIV. For example, if the rule were looking at the TOTALINVOICES row for division 01, see Figure 77“Sample Data for DO EXTRACT Definition” on page 2-112, the variable DIV would be set to 01. The names of the next two variables extracted would be set to TAMT01 and TCOST01, substituting 01 for %%DIV in the variable names TAMT%%DIV and TCOST%%DIV.

Similarly, if the rule were looking at the TOTALINVOICES row for division 02, it would extract variables whose names would be DIV, equal to a value of 02, TAMT02, and TCOST02.

While the variable has been specified only once, no matter how many divisions there are in the file, the result will be unique variable names for the invoice amount and cost for each division.

This capability to have the system dynamically generate variable names is a powerful feature of CONTROL-M / Analyzer rule definition parameters, which is used several times within this sample rule.

Examining the CHKDETL Block

You will now examine the CHKDETL block to see what balancing actions are performed on each detail row of the file.

1. Using **PF08/PF20**, or execute the DOWN command, scroll forward until you see the ON block EXECUTE CHKDETL.

2. Place the cursor on the line beginning with the string “EXECUTE CHKDETL,” and press **PF08/PF20**, or execute the DOWN command.

Figure 79 ON Block EXECUTE CHKDETL

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN04
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| EXECUTE CHKDETL  UPON                                                   C   |
| ON DATA                                                                     |
| IF       DIV NE 01 AND DIV NE 02 AND DIV NE 03                          C   |
|   DO SET      = PROBLEM='DIV'                                           C   |
|   DO SET      = ERRORS=ERRORS+1                                         C   |
|   DO BLOCK    = PRINT    ARG PROBLEM                                    C   |
| IF       DIV = 01 AND AMT > 400000                                      C   |
|   DO SET      = PROBLEM='AMT'                                           C   |
|   DO SET      = ERRORS=ERRORS+1                                         C   |
|   DO BLOCK    = PRINT    ARG PROBLEM                                    C   |
| IF       DIV = 01 AND AMT > 200000 AND AMT <= 400000                    C   |
|   DO SET      = PROBLEM='AMT-WRN'                                       C   |
|   DO SET      = TOLERABLES=TOLERABLES+1                                 C   |
|   DO BLOCK    = PRINT    ARG PROBLEM                                    C   |
| IF       DIV = 02 OR DIV = 03 AND AMT > 200000                          C   |
|   DO SET      = PROBLEM='AMT'                                           C   |
|   DO SET      = ERRORS=ERRORS+1                                         C   |
|   DO BLOCK    = PRINT    ARG PROBLEM                                    C   |
| IF       DIV = 02 OR DIV = 03 AND AMT > 100000 AND AMT <= 200000        C   |
|   DO SET      = PROBLEM='AMT-WRN'                                       C   |
PLEASE FILL IN RULE DEFINITION.                                       15.31.20
```

The parameters of the rule are defined as follows:

- The ON DATA block checks each field extracted from the detail line, according to the user criteria as shown in “Sources of Data—Reports and Files” on page 2-72. A series of IF statements determines if the balancing criteria are met.

  In the first DO SET parameter for each IF statement, if there is a problem, a variable called PROBLEM is set to equal a code that describes that problem. For example, in the first IF parameter, the code DIV is set if the division does not equal 01, 02, or 03.

- In the second DO SET parameter for the first, second, and fourth IF statements that you see on the screen, the error counter, ERRORS, is incremented by 1.

  In the third IF condition, another counter, TOLERABLES, is incremented and the variable, ERRORS, is not incremented. This occurs when the IF condition encounters an “unusual” situation, but not an error. The data, in that case, would be considered OK.

- After the first two DO SET statements in each IF statement, the rule then executes an ON block called PRINT, using the DO BLOCK parameter. The DO BLOCK action parameter enables you to

  — group instructions in a balancing rule together, organizing a more complex rule into logical sections

  — reference an ON block, such as PRINT, from several other ON blocks in the rule

Without DO BLOCK actions, ON blocks are executed sequentially. ON blocks that are called using DO BLOCK parameters should therefore be located after the last sequential ON block in the rule. This ensures that these ON blocks are not executed except when called by a DO BLOCK parameter.

3 Examine the rest of this ON block. Use **PF08/PF20**, or execute the DOWN command, and scroll forward to the bottom of the rule.

Figure 80 End of Rule ARWARN04

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN04
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| IF       DIV = 02 OR DIV = 03 AND AMT > 100000 AND AMT <= 200000        C   |
|   DO SET      = PROBLEM='AMT-WRN'                                      C   |
|   DO SET      = TOLERABLES=TOLERABLES+1                                C   |
|   DO BLOCK    = PRINT    ARG PROBLEM                                   C   |
| IF       MONTHYR NE DB_PROCESS_MONTH                                   C   |
|   DO SET      = PROBLEM='DATE'                                         C   |
|   DO SET      = ERRORS=ERRORS+1                                        C   |
|   DO BLOCK    = PRINT    ARG PROBLEM                                   C   |
| ========================================================================== |
| EXECUTE PRINT    UPON                                                   C   |
| ON DATA                                                                     |
| ALWAYS                                                                      |
|   DO PRINT    = %%DIV %%NAME %%CUSTNO %%INVNO %%DATE %%AMT %%PROBLE F   C Y |
|             M                                                               |
| ========================================================================== |
| EXECUTE          UPON                                                   C   |
| ON                                                                          |
======= >>>>>>>>>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<<<<< =====


PLEASE FILL IN RULE DEFINITION.                                      15.33.09
```

The EXECUTE PRINT block has just one parameter, a DO PRINT instruction. Rather than having a separate ON block such as this one for printing, the rule also could have been defined with multiple instances of these same print instructions in the CHKDETL block. This illustrates how rules can often be written differently and still accomplish the same balancing task.

Ending the rule

You will now examine the ON block, ENDJOB, which is executed sequentially after all file input has been processed by the ON block called INPUT. It is physically located further up in this rule.

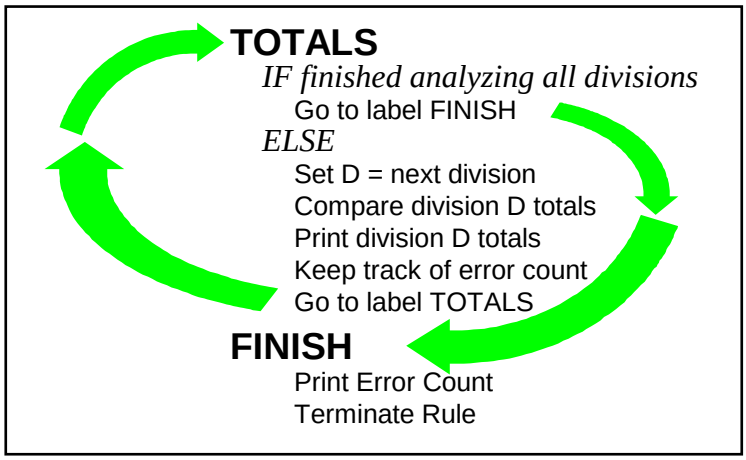
The last tasks before this rule ends are to

- compare divisional totals accumulated from the detail rows for each division to the divisional totals extracted from the file
- report these totals with any errors
- terminate the rule

To accomplish this, only one generic set of balancing instructions is necessary for all divisions. This is done by generalizing the instructions in order to insert the division code into the variable names. This requires using the %% operator, as explained in “DO Parameters” on page 2-112.

The flow of logic in the ON block, ENDJOB, is illustrated in the following diagram:

Figure 81 Flow of ON Block Logic



- A section of the ON block, called TOTALS, continually checks if all division totals are analyzed.
- If all division totals are analyzed, the rule proceeds to another section of the ON block, called FINISH, where the final error count is printed and the rule is terminated.
- If not (ELSE), a variable, *D*, is read, containing the current division name attached to the variable names. Division *D* totals are analyzed and printed. After the analysis, the rule returns to the section of the ON block called TOTALS.

1 Now examine the actual rule to see how this logic is defined. Use **PF07/PF19**, or execute the UP command, to scroll backward until you see ON block EXECUTE ENDJOB on the screen.

2 Place the cursor on the line beginning with the string “EXECUTE ENDJOB,” and press **PF08/PF20**, or execute the DOWN command.

Figure 82 ON Block EXECUTE ENDJOB

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN04
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| EXECUTE ENDJOB   UPON                                                    C   |
| ON DATA                                                                      |
| ALWAYS                                                                       |
|   DO SET      = I=1                                                      C   |
| LABEL: TOTALS                                                                |
| IF       I GT 3                                                          C   |
|   DO GOTO     LABEL FINISH                                                   |
| ELSE                                                                         |
|   DO SET      = D=0%%I                                                   C   |
|   DO SET      = DB_INVAMT%%D=AMT%%D                                      C   |
|   DO SET      = DB_COST%%D=COST%%D                                       C   |
|   DO SET      = PROBLEM=' '                                              C   |
|   DO SET      = PROB1=' '                                                C   |
| IF       AMT%%D NE TAMT%%D                                               C   |
|   DO SET      = PROBLEM='    TOTAMT'                                     C   |
|   DO SET      = ERRORS=ERRORS+1                                          C   |
| IF       COST%%D NE TCOST%%D                                             C   |
|   DO SET      = PROB1='    TOTCOST'                                      C   |
|   DO SET      = ERRORS=ERRORS+1                                          C   |
| ALWAYS                                                                       |
PLEASE FILL IN RULE DEFINITION.                                        15.37.18
```

This portion of the ON block does the following:

1. It starts creating a variable, *I*, which represents the division number being processed (either 1, 2, or 3).

   There are exactly 3 divisions to process. If *I* is larger than 3, all divisions have been processed and the rule can proceed to the section called FINISH.

2. The ON block creates a second variable *D*, which represents the division code that is attached to the variable names (either 01, 02, or 03).

   When a variable name is prefixed with the string DB_, CONTROL-M / Analyzer identifies the variable as a database variable. In this rule, variables are updated only if the rule terminates OK. The value of two database variables is set for each division:

   `DB_INVAMTxx` and `DB_COSTxx`

where xx is the division code. However, until the block performs a “commit” operation on these variables, these values have not been changed in the database.

3. Two variables, PROBLEM and PROB1, are set to equal a blank value. These variables are used to hold error conditions, if the totals are out of balance.

4. The ON block then performs the balancing analysis to see if the invoice amount for the division value that is set for AMT%%D (which was accumulated in the ON block CHKDETL) is equal to the total invoice amount, TAMT%%D (which, in turn, was extracted from the TOTALS line in the file).

   If these two amounts are not equal, the value set for the variable PROBLEM is set to the error message you want on the print line, replacing its previous blank value.

- The error count is then updated.

3 Press **PF08/PF20**, or execute the DOWN command, to scroll forward once.

Figure 83 Analysis on Cost Amounts

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN04
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| IF       COST%%D NE TCOST%%D                                          C    |
|   DO SET      = PROB1='    TOTCOST'                                  C    |
|   DO SET      = ERRORS=ERRORS+1                                      C    |
| ALWAYS                                                                     |
|   DO PRINT    = %%BLANK                                          F   C    |
|   DO PRINT    = DIVISION TOTAL CHECK                             F   C    |
|   DO PRINT    = %%D SALES: CALC  %%AMT%%D                        F   C    |
|   DO PRINT    = %%D SALES: FILE %%TAMT%%D %%PROBLEM              F   C    |
|   DO PRINT    = %%D  COST: CALC  %%COST%%D                       F   C    |
|   DO PRINT    = %%D  COST: FILE %%TCOST%%D %%PROB1               F   C    |
|   DO PRINT    = %%BLANK                                          F   C    |
|   DO SET      = I=INT(I+1)                                           C    |
|   DO GOTO     LABEL TOTALS                                               |
| LABEL: FINISH                                                              |
| ALWAYS                                                                     |
|   DO PRINT    = %%BLANK                                          F   C    |
|   DO PRINT    = ERRORS %%ERRORS  TOLERABLE ERRORS %%TOLERABLES   F   C    |
| IF       ERRORS > 0                                                    C    |
|   DO COMMIT   = NONE     VAR                                          C    |
|   DO TERMINAT = NOTOK    COD 0999                                           |
PLEASE FILL IN RULE DEFINITION.                                      15.38.37
```

As you continue looking through the rule, you see the same analysis that was done for problem analysis is performed on the cost amounts.

1. Compare the accumulated total cost for each division to the total cost amount that was extracted from the file.

2. Print the calculated and extracted amounts with appropriate headings and associated error messages.

4 Now that analysis of this division's totals has been completed, add 1 to the division count.

Return to the beginning of the ON block, to the section called TOTAL, where the rule checks if analysis has been completed for all divisions.

5 Look at the last few instructions in the rule. Place the cursor on the line beginning with the character string “LABEL: FINISH,” and press **PF08/PF20**, or execute the DOWN command.

Figure 84 LABEL: FINISH

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN04
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| LABEL: FINISH                                                               |
| ALWAYS                                                                      |
|   DO PRINT    = %%BLANK                                              F   C  |
|   DO PRINT    = ERRORS %%ERRORS  TOLERABLE ERRORS %%TOLERABLES       F   C  |
| IF       ERRORS > 0                                                      C  |
|   DO COMMIT   = NONE     VAR                                              C  |
|   DO TERMINAT = NOTOK    COD 0999                                            |
| IF       ERRORS = 0                                                      C  |
|   DO COMMIT   = ALL      VAR                                              C  |
|   DO PRINT    = DIVISIONAL FILES ARE OK. DATA BASE IS UPDATED        F   C  |
| IF       TOLERABLES > 0                                                  C  |
|   DO TERMINAT = OK       COD 0009                                            |
| IF       ERRORS = 0 AND TOLERABLES = 0                                   C  |
|   DO TERMINAT = OK       COD 0000                                            |
| ========================================================================== |
| EXECUTE CHKDETL  UPON                                                    C  |
| ON DATA                                                                     |
| IF       DIV NE 01 AND DIV NE 02 AND DIV NE 03                           C  |
|   DO SET      = PROBLEM='DIV'                                            C  |
|   DO SET      = ERRORS=ERRORS+1                                          C  |
PLEASE FILL IN RULE DEFINITION.                                        15.39.49
```

The LABEL: FINISH instructions

1. print the final counts of errors and tolerable errors that have been accumulated throughout the execution of the rule

2. do not commit, or update, the database variables if there were any serious errors encountered at any time during the rule execution

   In such cases the rule terminates NOTOK, with a user code of 999.

3. commit all variables, meaning that CONTROL-M / Analyzer updates the database at this time, if there were no serious errors

   The rule then prints a confirming message that the database has been updated. The database variables that are updated are INVAMT01, INVAMT02, INVAMT03, COST01, COST02, and COST03.

4. terminates the rule with user code 9 if there are tolerable errors

5. terminates the rule with a user code of 0 if there are no errors

Conclusion

This concludes your examination of the rule definition. You will now take a brief look at two user reports that result as output for this rule—one where the rule executed NOTOK, and one where the rule executed OK.

There are two jobs provided in the sample JCL library, related to rule ARWARN04: RUNRULE4 and RUNRUL4A.

Job RUNRULE4

RUNRULE4 uses an input file that contains serious errors. You may submit this job and review the outputs.

Figure 85 SYSUSER Report from RUNRULE4

```
         INTRAC DAILY DIVISIONAL INVOICE ANALYSIS
                     FOR  000904
-- ------- -------- ------- ------- --------- ---------
DV CUST    CUST#    INV #   DATE    AMOUNT    PROBLEM
-- ------- -------- ------- ------- --------- ---------
01 TERRANO 1TER001 CJ39398 030800   945245.00 AMT
02 SWANSON 2SWA001 MJ24545 040700    10987.55 DATE
02 ALBANAN 2ALB001 MJ24563 040800   110300.00 AMT-WRN
02 TRANKLE 2TRA002 BJ24572 040800      500.00 DATE
04 DURKENS 3DUR001 ZZ10906 040800   275000.00 DIV

DIVISION TOTAL CHECK
01 SALES: CALC  978365.250
01 SALES: FILE  978365.25
01  COST: CALC  938455.000
01  COST: FILE  901000.00

DIVISION TOTAL CHECK
02 SALES: CALC  679025.520
02 SALES: FILE  679025.52
02  COST: CALC  541873.000
02  COST: FILE  541873.00

DIVISION TOTAL CHECK
03 SALES: CALC  293764.810
03 SALES: FILE  293564.81
03  COST: CALC  264202.000
03  COST: FILE  264202.00

ERRORS 6.00  TOLERABLE ERRORS 1.000
```

In this report, because there were serious errors resulting in condition NOTOK, no database update occurred. There were several problems with the input file, including:

- amounts that were intolerable
- an invoice date with the wrong month
- an invoice date with the wrong year
- a record with a division code of 04.

One tolerable error produced a warning message on the amount.

When checking divisional totals in the ENDJOB block, two more serious errors were found—calculated total cost for division 01, and total invoice amount for division 03, were not in agreement with the file amounts.

Job RUNRUL4A

RUNRUL4A is almost identical to RUNRULE4 except that it uses a file with no serious error conditions.

6 You may submit this job now and review the output.

Figure 86 SYSUSER Report from RUNRUL4A

```
        INTRAC DAILY DIVISIONAL INVOICE ANALYSIS
                     FOR  000904
-- ------- -------- ------- ------- --------- ---------
DV CUST    CUST#    INV #   DATE    AMOUNT    PROBLEM
-- ------- -------- ------- ------- --------- ---------
01 TERRANO 1TER001 CJ39398 030800   345245.00 AMT-WRN
02 ALBANAN 2ALB001 MJ24563 040800   110300.00 AMT-WRN

DIVISION TOTAL CHECK
01 SALES: CALC  378365.250
01 SALES: FILE  378365.25
01  COST: CALC  338455.000
01  COST: FILE  338455.00

DIVISION TOTAL CHECK
02 SALES: CALC  679025.520
02 SALES: FILE  679025.52
02  COST: CALC  541873.000
02  COST: FILE  541873.00

DIVISION TOTAL CHECK
03 SALES: CALC  368764.810
03 SALES: FILE  368764.81
03  COST: CALC  319302.000
03  COST: FILE  319302.00

ERRORS 0  TOLERABLE ERRORS 2.000
DIVISIONAL FILES ARE OK. DATA BASE IS UPDATED
```

This report, from RUNRUL4A, indicates no serious errors, although you did receive two warning messages.

As noted, the file is declared OK and the database has been updated.

7 You can now exit this rule and proceed to the next problem.

Problem 5: Check Weekly Database Amounts

One of the INTRAC managers just said, “Now our invoices are in balance. But you have a problem with profits. Every month one of our divisions is doing poorly.” The manager wondered if CONTROL-M / Analyzer could be used, not just as a data quality control tool, but also as an early warning system to management.

Problem

Each day, as the invoices from each division are processed, the CONTROL-M / Analyzer database is updated with divisional totals for invoice amounts and costs. However, even when daily invoice amounts are balanced and meet the daily criteria, there can be problems that show up only over a period of time. For example, it may be valid for a division to have a very small amount invoiced on a given day. However, if this situation continued for a week, this would be very abnormal.

INTRAC has two problems in this situation:

- No one at INTRAC has ever defined weekly “reasonableness” criteria for invoice and cost amounts.
- The INTRAC executives suspect that from time to time there are problems with both the volume, or total dollar amount, of invoices, and with the profit margins of each of the divisions.

Solution — Overview

The head of ISD has met with the executives and determined the following guidelines:

- Weekly sales for Division 01 must be at least $2,000,000.
- Weekly sales for Division 02 must be at least $3,000,000.
- Weekly sales for Division 03 must be at least $600,000.
- Weekly profit margin for Division 01 must be at least 15%.
- Weekly profit margin for Division 02 must be at least 20%.
- Weekly profit margin for Division 03 must be at least 38%.
- Profit margin = (Sales - Cost of Sales) * 100 / Sales

Since these are new guidelines being tried for the first time, you have been asked to ensure that they are easy to change.

Solution – Details

In this CONTROL-M / Analyzer solution, the rule

- accumulates weekly sales and costs, by division, by retrieving these amounts from various generations in the database
- calculates margins for each division
- analyzes weekly sales and margins by division, and reports these amounts and any abnormal situations
- ends OK or NOTOK, depending on errors found

In this rule, you will learn how to refer to and access various *generations* of data from the database.

Opening Rule ARWARN05

1 Select rule ARWARN05 for browsing.

If you are not familiar with how to select a rule for browsing, review the instructions at the beginning of the discussion of rule ARWARN02 in Problem 2.

The rule consists of the following three blocks:

- INIT – Sets initial values for the variables that accumulate weekly totals. Also initializes an error counter and a blank, for printing blank lines.
- ACCUMSLS – Accumulates weekly sales and cost amounts and calculates margins for each division.
- CHECK – Checks the sales and margins for each division, prints the data with relevant problem messages, and terminates the rule accordingly.

Figure 87 Rule ARWARN05

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN05
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| OWNER   M66                      GROUP INTRAC                                |
| UPDATED 11/09/99  -  11:53:49    BY M66                                      |
| DESC    CHECK WEEKLY SALES AND COST OF SALES AMOUNTS                         |
| OPTIONS                                                                      |
| ============================================================================ |
| EXECUTE INIT     UPON                                                     C   |
| ON DATA                                                                      |
| LABEL: SETVARS                                                               |
| ALWAYS                                                                       |
|   DO SET      = WEEKSALES01=0                                            C   |
|   DO SET      = WEEKSALES02=0                                            C   |
|   DO SET      = WEEKSALES03=0                                            C   |
|   DO SET      = WEEKCOST01=0                                             C   |
|   DO SET      = WEEKCOST02=0                                             C   |
|   DO SET      = WEEKCOST03=0                                             C   |
|   DO SET      = CTR1=0                                                   C   |
|   DO SET      = BLANK=' '                                                C   |
| ============================================================================ |
| EXECUTE ACCUMSLS UPON                                                    C   |
| ON DATA                                                                      |
PLEASE FILL IN RULE DEFINITION.                                        15.43.46
```

2 In the first ON block, look at the LABEL: SETVARS section; note the first six variables that are initialized with DO SET actions—from WEEKSALES01 through WEEKCOST3. These variables are used for accumulating weekly invoice and cost amounts.

3 Now examine the second ON block, called ACCUMSLS. Place the cursor in the COMMAND field and press **PF08/PF20**, or execute the DOWN command.

Figure 88 LABEL ACCUM in the ACCUMSLS ON Block

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN05
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| LABEL: ACCUM                                                                |
| ALWAYS                                                                      |
|   DO SET      = WEEKSALES01=WEEKSALES01 + DB_INVAMT01@G%%CTR1           C   |
|   DO SET      = WEEKSALES02=WEEKSALES02 + DB_INVAMT02@G%%CTR1           C   |
|   DO SET      = WEEKSALES03=WEEKSALES03 + DB_INVAMT03@G%%CTR1           C   |
|   DO SET      = WEEKCOST01=WEEKCOST01 + DB_COST01@G%%CTR1               C   |
|   DO SET      = WEEKCOST02=WEEKCOST02 + DB_COST02@G%%CTR1               C   |
|   DO SET      = WEEKCOST03=WEEKCOST03 + DB_COST03@G%%CTR1               C   |
|   DO SET      = CTR1=INT(CTR1+1)                                        C   |
| IF       CTR1<5                                                         C   |
|   DO GOTO     LABEL ACCUM                                                   |
| ========================================================================== |
| EXECUTE CHECK    UPON                                                     C   |
| ON DATA                                                                     |
| ALWAYS                                                                      |
|   DO SET      = MARGIN1=(WEEKSALES01-WEEKCOST01)/WEEKSALES01            C   |
|   DO SET      = MARGIN2=(WEEKSALES02-WEEKCOST02)/WEEKSALES02            C   |
|   DO SET      = MARGIN3=(WEEKSALES03-WEEKCOST03)/WEEKSALES03            C   |
| IF       WEEKSALES01 < 2000000                                          C   |
|   DO SET      = PROBLEM01='SALES BELOW TARGET'                          C   |
PLEASE FILL IN RULE DEFINITION.                                        15.44.57
```

4 Look at the first DO SET instruction in this ON block.

It sets variable WEEKSALES01 equal to the sum of the variables WEEKSALES01 and DB_INVAMT01@G%%CTR1.

When the rule started, variable WEEKSALES01 was initialized to 0.

The expression DB_INVAMT01@G%%CTR1 is defined as follows:

- The string DB_INVAMT01 indicates that the variable INVAMT01 in the database is being referenced.
- The @G symbol indicates a specific generation of variable INVAMT01 is being referenced. The string %%CTR1 is the generation of the variable being referred to by the @G symbol.

  When the rule started, %%CTR1 was set to 0.
- The expression DB_INVAMT01@G%%CTR1, therefore, refers to the value of generation 0 of database variable INVAMT01.

Each day, as INVAMT01 is updated from the daily invoice total for Division 1, as set forth in the explanation of rule ARWARN04 in the section describing Problem 4, a new generation of INVAMT01 is created. The most recent generation of INVAMT01 is always generation 0. The next most recent generations are generations 1, 2, and so on.

5 Now examine at the entire ACCUM procedure, shown in Figure 88 on page 2-128. The rule selects the five most recent generations of each of the database variables, representing the most recent data for the week, as follows:

1. The rule first adds the value of generation 0 of each of the database variables into WEEKSALES01, WEEKSALES02, and so on.

2. The rule then adds 1 to the counter, CTR1, which represents the generation number.

3. If CTR1 is less than 5, it returns to LABEL: ACCUM to calculate another day's worth of values. If CTR1 is greater than or equal to 5, meaning that all the figures for the week have been accumulated, the rule proceeds to accumulate margins based on these figures.

4. After calculating margins, the rule begins the analysis on all the figures.

Examining the Balancing Checks of the ON Block EXECUTE ACCUMSLS

6 Place the cursor on the line beginning with the string "IF WEEKSALES01=," and press **PF08/PF20**, or enter the DOWN command.

Figure 89 Balancing Checks of the ON Block EXECUTE ACCUMSLS

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN05
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| IF       WEEKSALES01 < 2000000                                          C   |
|   DO SET      = PROBLEM01='SALES BELOW TARGET'                          C   |
|   DO SET      = CTR1=999                                                C   |
| ELSE                                                                        |
|   DO SET      = PROBLEM01=BLANK                                         C   |
| IF       WEEKSALES02 < 3000000                                          C   |
|   DO SET      = PROBLEM02='SALES BELOW TARGET'                          C   |
|   DO SET      = CTR1=999                                                C   |
| ELSE                                                                        |
|   DO SET      = PROBLEM02=BLANK                                         C   |
| IF       WEEKSALES03 <  600000                                          C   |
|   DO SET      = PROBLEM03='SALES BELOW TARGET'                          C   |
|   DO SET      = CTR1=999                                                C   |
| ELSE                                                                        |
|   DO SET      = PROBLEM03=BLANK                                         C   |
| IF       MARGIN1 < 0.15                                                 C   |
|   DO SET      = PROBLEM04='MARGIN BELOW TARGET'                         C   |
|   DO SET      = CTR1=999                                                C   |
| ELSE                                                                        |
|   DO SET      = PROBLEM04=BLANK                                         C   |
PLEASE FILL IN RULE DEFINITION.                                        15.46.10
```

This group of IF statements performs the balancing checks for the accumulated amounts, in the following basic pattern:

1. Tests the accumulated amount to see if it breaks the rule. For example, the first IF statement is IF WEEKSALES01 < 2000000.

2. If the amount is in error, that is, if the first IF statement is less than 2000000

   — set an error message for later printing, in the first case “SALES BELOW TARGET”, and

   — set the variable CTR1 to 999

   999 in all these IF statements indicates that one or more of the rules has been broken. Later, this will be used to terminate the rule NOTOK.

3. If the test is OK, that is, if it is greater than the target, set the error message to blank.

7 Press **PF08/PF20**, or enter the DOWN command, to scroll forward one screen to examine the remainder of these balancing checks.

Figure 90 Last Balancing Checks of the ON Block EXECUTE ACCUMSL

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN05
COMMAND ===>                                                     SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
|   DO SET      = PROBLEM03=BLANK                                        C    |
| IF       MARGIN1 < 0.15                                                C    |
|   DO SET      = PROBLEM04='MARGIN BELOW TARGET'                        C    |
|   DO SET      = CTR1=999                                               C    |
| ELSE                                                                        |
|   DO SET      = PROBLEM04=BLANK                                        C    |
| IF       MARGIN2 < 0.2                                                 C    |
|   DO SET      = PROBLEM05='MARGIN BELOW TARGET'                        C    |
|   DO SET      = CTR1=999                                               C    |
| ELSE                                                                        |
|   DO SET      = PROBLEM05=BLANK                                        C    |
| IF       MARGIN3 < 0.38                                                C    |
|   DO SET      = PROBLEM06='MARGIN BELOW TARGET'                        C    |
|   DO SET      = CTR1=999                                               C    |
| ELSE                                                                        |
|   DO SET      = PROBLEM06=BLANK                                        C    |
| ALWAYS                                                                      |
|   DO PRINT    = INTRAC WEEKLY INVOICE ANALYSIS                       F   C    |
|   DO PRINT    =         FOR %%SYSDATE                                F   C    |
|   DO PRINT    = %%BLANK                                              F   C    |
PLEASE FILL IN RULE DEFINITION.                                          15.47.07
```

8 After examining all the balancing checks, scroll forward once more to the end of this rule by pressing **PF08/PF20**, or entering the DOWN command.

Figure 91 End of Rule ARWARN05

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN05
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| ELSE                                                                         |
|   DO SET      = PROBLEM06=BLANK                                          C   |
| ALWAYS                                                                       |
|   DO PRINT    = INTRAC WEEKLY INVOICE ANALYSIS                       F   C   |
|   DO PRINT    =         FOR %%SYSDATE                                F   C   |
|   DO PRINT    = %%BLANK                                              F   C   |
|   DO PRINT    = SALES OF DIV 1 - %%WEEKSALES01 %%PROBLEM01           F   C   |
|   DO PRINT    = SALES OF DIV 2 - %%WEEKSALES02 %%PROBLEM02           F   C   |
|   DO PRINT    = SALES OF DIV 3 - %%WEEKSALES03 %%PROBLEM03           F   C   |
|   DO PRINT    = %%BLANK                                              F   C   |
|   DO PRINT    = MARGIN OF DIV 1 - %%MARGIN1 %%PROBLEM04              F   C   |
|   DO PRINT    = MARGIN OF DIV 2 - %%MARGIN2 %%PROBLEM05              F   C   |
|   DO PRINT    = MARGIN OF DIV 3 - %%MARGIN3 %%PROBLEM06              F   C   |
| IF       CTR1=999                                                            C   |
|   DO TERMINAT = NOTOK    COD 0099                                             |
| ELSE                                                                         |
|   DO TERMINAT = OK       COD 0000                                             |
| ============================================================================ |
| EXECUTE          UPON                                                        C   |
| ON                                                                           |
PLEASE FILL IN RULE DEFINITION.                                        15.47.56
```

After the rule performs all checks, it prints the balancing report. Note how any problems that are encountered are referred to by using the variable names PROBLEM01, PROBLEM02, and so on, in each print line.

The rule terminates either OK or NOTOK, depending on the value of CTR1. As defined earlier, if any errors are encountered, variable CTR1 is set to 999.

Conclusion

1 You may now submit job RUNRULE5 and review the SYSUSER report.

Figure 92 SYSUSER Report from RUNRULE5

```
INTRAC WEEKLY INVOICE ANALYSIS
        FOR  000924

SALES OF DIV 1 - 1659125.250 SALES BELOW TARGET
SALES OF DIV 2 - 3110109.520
SALES OF DIV 3 - 33589357.810

MARGIN OF DIV 1 - -0.114 MARGIN BELOW TARGET
MARGIN OF DIV 2 - 0.202
MARGIN OF DIV 3 - 0.642
```

The figures in the report which you produce may vary, depending on how often the demonstration rules have been run in your installation.

2 You can now exit this rule and proceed to the next problem.

Problem 6: Balance Monthly Commissions

The INTRAC ISD manager has asked for one last balancing job to be performed—a detailed review of the monthly commissions report.

Problem

Every month, the payroll department receives many adjustments from the sales operations department relating to salesperson commissions, as referred to in Problem 2. Salespeople and sales managers have been complaining to the executives of INTRAC that they do not trust the detailed sales commissions report. They believe that there may be several problems, and feel that

- detailed invoice amounts do not equal the total reported sales
- reported commissions owed do not equal total sales multiplied by a commission percentage
- the reported payroll amount, which is the amount to be paid this month, does not equal YTD commissions owed minus YTD commissions paid

Unlike the report that was prepared in Problem 2, the sales management needs a balancing report that shows salesperson names. They need this report to review it with their salespeople.

Solution — Overview

The payroll department will employ a CONTROL-M/Analyzer rule to audit every line of the Sales Commission Detail report, for the purpose of checking the problems reported above. They have designed a report that will summarize the reported amounts and calculated amounts. Having reviewed the report with the sales managers and received agreement from them, you are ready to write the rule.

Solution – Details

The rule consists of the following four ON blocks:

- INIT – initializes values of variables which are used in the rule
- INPUT – extracts data from the report
- CHKNAME – establishes a table of salesperson numbers
- FINISH – analyzes all the data in a table, prints the balancing report, and terminates the job

In this rule, you will

- use the STOP and AND parameters of a WHEN clause to force CONTROL-M / Analyzer to look for combinations of conditions in a report, before extracting data

- set up and process a table of variables

Examine this rule online. Proceed to the Rule Definition facility, to the Rule List screen.

1 Select rule ARWARN06 for browsing.

If you are not familiar with how to select a rule for browsing, review the instructions at the beginning of the discussion of rule ARWARN02 in Problem 2.

Figure 93 Rule ARWARN06

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN06
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| OWNER   M66                      GROUP INTRAC                                |
| UPDATED 03/04/99  -  15:30:05    BY M66                                      |
| DESC    CHECK MONTH SALES COMMISSIONS REPORT                                 |
| OPTIONS                                                                      |
| ============================================================================ |
| EXECUTE INIT     UPON                                                     C   |
| ON DATA                                                                      |
| ALWAYS                                                                       |
|   DO SET      = TOTPEOPLE=0                                               C   |
|   DO SET      = LASTSALES=' '                                             C   |
|   DO SET      = ERRORCOUNT=0                                              C   |
|   DO SET      = BLANK=' '                                                 C   |
| ============================================================================ |
| EXECUTE INPUT    UPON                                                     C   |
| ON SYSOUT     PROCST          PGMST STEP01   DDNAME SYSUT2   JOBNM           |
|    MODE PG    LINECT 0000    DATASTAMP                                       |
| WHEN LINE 001     - 005     COL 002     - 014              STOP Y  AND/OR A |
|      STRING = SALESPERSON                                                    |
| STOP LINE 001     - 002     COL 001     - 020                               |
|      STRING = INTRAC                                                         |
PLEASE FILL IN RULE DEFINITION.                                        15.50.49
```

In the INIT ON block, two variables, TOTPEOPLE and LASTSALES, process a table containing each salesperson's relevant amounts from the report. These variables are discussed later in these instructions.

In the next ON block, INPUT extracts data from the report.

2 Place the cursor on the line beginning with the character string “EXECUTE INPUT,” and press **PF08/PF20**, or enter the DOWN command.

Figure 94 EXECUTE INPUT ON Block

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN06
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| EXECUTE INPUT    UPON                                                    C   |
| ON SYSOUT     PROCST          PGMST STEP01   DDNAME SYSUT2   JOBNM          |
|    MODE PG    LINECT 0000    DATASTAMP                                      |
| WHEN LINE 001     - 005     COL 002     - 014              STOP Y  AND/OR A |
|      STRING = SALESPERSON                                                   |
| STOP LINE 001     - 002     COL 001     - 020                              |
|      STRING = INTRAC                                                        |
| WHEN LINE L01+001 - 060     COL 050     - 060              STOP    AND/OR   |
|      STRING = /                                                             |
|   DO EXTRACT  = SALESNO                                                     |
|                 LEVEL 1 LINE +000 COL 0035 - 0037 PROCESS          TYP      |
|   DO EXTRACT  = AMOUNT                                                      |
|                 LEVEL 0 LINE +000 COL 0068 - 0078 PROCESS          TYP      |
|   DO EXTRACT  = SUM1%%SALESNO                                               |
|                 LEVEL 0 LINE +000 COL 0068 - 0078 PROCESS          TYP SU   |
|   DO BLOCK    = CHKNAME  ARG                                            C   |
| WHEN LINE 005     - 005     COL 002     - 014              STOP Y  AND/OR A |
|      STRING = SALESPERSON                                                   |
| STOP LINE 001     - 002     COL 001     - 020                              |
|      STRING = INTRAC                                                        |
PLEASE FILL IN RULE DEFINITION.                                       15.51.58
```

Compound WHEN Statements

This ON block uses compound WHEN statements to locate certain sections of the report. The first compound WHEN statement performs the following actions:

1. It locates the string “SALESPERSON.”

2. Whenever this string is found, the second WHEN statement searches for the string “ / ” on every subsequent line.

3. Every time the string “ / ” is found, the DO actions are performed. Because for third DO action the TYPE parameter is set to SU, the rule automatically totals the amount field found in columns 68 through 78, into a variable called SUM1%%SALESNO.

4. The DO actions are stopped. The ON block searches again for the string, “SALESPERSON.” When it finds the string, “INTRAC,” in lines 1 through 2, this indicates a new page of the report.

Why was a compound WHEN structure used for this operation?

To access the different sections of the report from which to extract data, the rule needs to uniquely identify those sections. For example, the rule may be searching for the string “ / ” but there may be sections of the report where you do not want to extract any data. This can be done using WHEN statements, which enable the rule to easily extract data from different areas of the report, without requiring that each section have a fixed number of lines.

Examine how this is done by looking at a sample page from the report:

Figure 95 Sample Page of the Sales Operations Department Report

```
INTRAC INC.                SALES OPERATIONS DEPARTMENT     DATE 09/12/00 TIME 23:50     PAGE  1

            S A L E S     C O M M I S S I O N S    D E T A I L          FROM 01/01/00  TO  01/12/00
| SALESPERSON   HARVEY J. JONES (197)  | INVOICE # | INVOICE DATE |   AMOUNT   | STATUS
+-------------------------------------------------------------------------------------------------
--------
|                                      | CJ39D94   | 02/02/00     |   2040.00 |COLLECTION   |
|                                      | CJ98443   | 03/03/00     |    245.00 |             |
|                                      | MJ24545   | 04/04/00     |  10344.40 |             |
|                                      | RJ10030   | 01/02/00     |   5329.39 |             |
|                                      | RJ12233   | 02/02/00     |   5000.00 |             |
|                                      | RJ30021   | 03/02/00     |  16670.10 |             |
|                                      | RJ32991   | 02/03/00     |  30300.44 |             |
|                                      | RJ33939   | 03/03/00     |    900.25 |             |
|                                      | RJ38001   | 04/04/00     |   1000.23 |             |
|                                      | RJ93043   | 04/04/00     |    415.89 |             |
|+------------------------------------------------------------------------------------------------
---------
  TOTAL SALES - THIS MONTH                                                 72245.70 | PAID IN FULL|
  ADJUSTMENTS                                                              -5800.70
  YTD SALES (INCLUDING THIS MONTH)                                       -127879.00
  YTD COMMISSIONS PAID                                                        0.00
  YTD COMMISSIONS OWED                                                     -5115.00
  COMMISSION PAYROLL AMOUNT                                                 5115.00
```

1. Find the line with the salesperson named “Harvey J. Jones,” and number “197.” Because this line was found as a result of the first WHEN parameter specified, this is called a LEVEL 1 line.

   Look at the DO EXTRACT statements shown in Figure 94 on page 2-134, and see how the first DO EXTRACT statement (LEVEL 1) is specified to extract the salesperson number (SALESNO). The next WHEN statement within a compound WHEN grouping would point to a LEVEL 2 line, and any other WHEN statements would point to LEVEL 3, LEVEL 4, and so on.

2. For the salesperson Harvey J. Jones, the rule processes all detail lines that are identified with a “ / ” between columns 50 through 60. Because these were found as a result of the second WHEN statement in the compound WHEN parameter, it is called a LEVEL 2 line.

NOTE

Because this WHEN statement is also the last WHEN statement in the compound WHEN grouping, it is also referred to as a LEVEL 0 line. LEVEL 0 always refers to the last WHEN statement specified within a compound WHEN grouping.

As defined in the compound WHEN grouping, the number of lines of data between the LEVEL 1 line (SALESNO) and the LEVEL 0 or LEVEL 2 (“/”) line may vary. That is, there can be any number of lines of data for Harvey J. Jones until the rule encounters the next SALESNO in the data it is searching.

Also, any other lines of the report that coincidentally contain a “/” character, such as the top two lines of the report, are not processed as detail lines for Harvey J. Jones because they are not preceded by a LEVEL 1 line containing the character string “SALESPERSON.”

3. The rule searches for LEVEL 0 lines until a line containing the string “INTRAC” is found (see the top of the sample report).

3 Now look at the next compound grouping of WHEN parameters. Place the cursor on the line beginning with the character string “WHEN LINE 005” and press **PF08/PF20**, or enter the DOWN command.

Figure 96 Second Compound Grouping of WHEN Parameters

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN06
COMMAND ===>                                                    SCROLL===> CRSR
+-------------------------------- BROWSE -----------------------------------+
| WHEN LINE 005     - 005     COL 002     - 014              STOP Y  AND/OR A |
|      STRING = SALESPERSON                                                  |
| STOP LINE 001     - 002     COL 001     - 020                             |
|      STRING = INTRAC                                                       |
| WHEN LINE 001     - 060     COL 001     - 020              STOP    AND/OR   |
|      STRING = TOTAL SALES                                                  |
|   DO EXTRACT  = SALESNO                                                    |
|                 LEVEL 1 LINE +000 COL 0035 - 0037 PROCESS          TYP     |
|   DO EXTRACT  = NAME%%SALESNO                                              |
|                 LEVEL 1 LINE +000 COL 0017 - 0033 PROCESS          TYP     |
|   DO EXTRACT  = MTHSLS%%SALESNO                                            |
|                 LEVEL 0 LINE +000 COL 0068 - 0078 PROCESS          TYP     |
|   DO EXTRACT  = YTDSLS%%SALESNO                                            |
|                 LEVEL 0 LINE +002 COL 0068 - 0078 PROCESS          TYP     |
|   DO EXTRACT  = YTDCOM%%SALESNO                                            |
|                 LEVEL 0 LINE +003 COL 0068 - 0078 PROCESS          TYP     |
|   DO EXTRACT  = YTDOWE%%SALESNO                                            |
|                 LEVEL 0 LINE +004 COL 0068 - 0078 PROCESS          TYP     |
|   DO EXTRACT  = PAYAMT%%SALESNO                                            |
|                 LEVEL 0 LINE +005 COL 0068 - 0078 PROCESS          TYP     |
PLEASE FILL IN RULE DEFINITION.                                      15.54.04
```

This compound WHEN grouping performs the following actions:

- The WHEN statement locates the string “SALESPERSON.”
- Whenever the string “SALESPERSON” is found, the WHEN statement searches on every subsequent line for the string, “TOTAL SALES.”

Every time the “TOTAL SALES” string is found, the WHEN statement performs the DO actions. Each time a value is extracted from the report, the salesperson number becomes part of the variable name. The result is a series of variables with names as follows:

```
--------------------------- Salesperson -------------------------
   1                          2                          3
NAME197                   NAME101                    NAME177
MTHSLS197                 MTHSLS101                  MTHSLS177
YTDSLS197                 YTDSLS101                  YTDSLS177
YTDCOM197                 YTDCOM101                  YTDCOM177
YTDOWE197                 YTD0WE101                  YTDOWE177
PAYAMT197                 PAYAMT101                  PAYAMT177
```

Each of these variables contains data extracted from the report.

- The DO actions are stopped. The ON block searches again for the string “SALESPERSON” when it finds the string INTRAC in lines 1 through 2, indicating a new page of the report.
- The ON block CHKNAME is called (see Figure 94 on page 2-134) for each detail line being processed. Now examine what the ON block CHKNAME does.

Examining the ON Block CHKNAME

4 Scroll forward to the bottom of the rule by pressing **PF08/PF20**, or by entering the DOWN command.

Figure 97 ON Block CHKNAME

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN06
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
|   DO PRINT    = ***** SERIOUS COMMISSION ERRORS *****                F   C   |
|   DO PRINT    = ***** TOTAL NUMBER OF ERRORS %%ERRORCOUNT *****      F   C   |
|   DO TERMINAT = NOTOK    COD 0999                                            |
| ELSE                                                                         |
|   DO PRINT    = %%BLANK                                              F   C   |
|   DO PRINT    = %%BLANK                                              F   C   |
|   DO TERMINAT = OK       COD 0000                                            |
| ============================================================================ |
| EXECUTE CHKNAME  UPON                                                    C   |
| ON DATA                                                                      |
| IF       SALESNO NE LASTSALES                                            C   |
|   DO SET      = TOTPEOPLE=INT(TOTPEOPLE+1)                               C   |
|   DO SET      = NUMBER%%TOTPEOPLE=SALESNO                                C   |
|   DO SET      = LASTSALES=SALESNO                                        C   |
| ============================================================================ |
| EXECUTE          UPON                                                    C   |
| ON                                                                           |
======= >>>>>>>>>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<<<<<< =====


PLEASE FILL IN RULE DEFINITION.                                       15.55.43
```

This ON block is only performed when the salesperson number has changed during processing of the rule. The variable, SALESNO, is the actual salesperson number that is extracted from the LEVEL 1 line on the report. LASTSALES is a variable that stores the salesperson number whenever processing begins for a new salesperson.

CHKNAME performs the following actions:

1. Adds 1 to the count of the number of salespeople, tracked in variable TOTPEOPLE.

2. Creates a table of salesperson numbers, which will be used later. The table, in its final form, will appear as follows:

```
Variable Name                Contents
  NUMBER1                      197
  NUMBER2                      101
  NUMBER3                      177
```

3. Because this is now a new salesperson being processed, set the variable LASTSALES equal to the new salesperson number.

Examining the ON Block FINISH

After processing the report, the rule proceeds to the next sequential ON block, FINISH, which contains most of the balancing parameters.

5 Scroll backward using **PF07/PF19**, or by entering the UP command, until you see the ON block FINISH on the screen.

6 Place the cursor on the line beginning with the character string “EXECUTE FINISH,” and press **PF08/PF20**, or enter the DOWN command.

Figure 98 ON Block FInish

```
        LIBRARY : CTB.PROD.RULES                                  RULE : ARWARN06
COMMAND ===>                                                      SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| EXECUTE FINISH   UPON                                                    C   |
| ON DATA                                                                      |
| ALWAYS                                                                       |
|   DO SET      = I=1                                                      C   |
|   DO PRINT    = TOTAL # OF SALESPEOPLE=%%TOTPEOPLE                   F   C   |
|   DO PRINT    = %%BLANK                                              F   C   |
| LABEL: TEST                                                                  |
| IF       I GT TOTPEOPLE                                                  C   |
|   DO GOTO     LABEL ENDJOB                                                   |
| ELSE                                                                         |
|   DO PRINT    = **** COMMISSION CONTROL - %%NAME%%NUMBER%%I ****     F   C   |
|   DO PRINT    = REPORTED SALES               %%MTHSLS%%NUMBER%%I     F   C   |
|   DO PRINT    = CALCULATED SALES                %%SUM1%%NUMBER%%I    F   C   |
|   DO PRINT    = ---------------------------------------------        F   C   |
|   DO SET      = DIFF1=ABS(SUM1%%NUMBER%%I-MTHSLS%%NUMBER%%I)             C   |
|   DO PRINT    = DIFFERENCE                           %%DIFF1         F   C   |
|   DO PRINT    = ---------------------------------------------        F   C   |
|   DO SET      = COMMCALC=YTDSLS%%NUMBER%%I*0.04                          C   |
|   DO PRINT    = REPORTED COMMISSIONS OWED    %%YTDOWE%%NUMBER%%I     F   C   |
|   DO PRINT    = CALCULATED COMMISSIONS OWED     %%COMMCALC           F   C   |
PLEASE FILL IN RULE DEFINITION.                                          15.56.18
```

The FINISH block is defined as follows:

- At the beginning of the FINISH block, variable “I” is set to the value 1. This variable keeps track of which salesperson's data, taken from the table of salesperson numbers created in CHKNAME, is being currently analyzed and reported.

- The section of this block called LABEL: TEST checks whether all salespeople have been processed.

  Because the report covers three salespeople in total, the variable TOTPEOPLE was set to 3 in the ON block CHKNAME. When the variable I, used to identify the salesperson on which you are currently working, is greater than 3, processing continues with the section in this ON block labelled ENDJOB.

  Otherwise, the rule proceeds to analyze and report on the current salesperson's data, which has been stored previously in the table of salesperson number.

- The remainder of the rule definition contains parameters that you learned in previous Problems.

Conclusion

With the next series of screens and report outputs (SYSUSER), you can compare the instructions in rule ARWARN06 to the output report produced.

Further explanation of rule definition parameters is available in the *CONTROL-M/Analyzer for z/OS User Guide.*

1 You may use the RUNRULE6 job in the JCL library to run this rule, which produces the following output:

Figure 99 Sample Output Produced by the RUNRULE6 Job

```
TOTAL # OF SALESPEOPLE=3

**** COMMISSION CONTROL -  HARVEY J. JONES  ****
REPORTED SALES                  72245.70
CALCULATED SALES                72245.700
-----------------------------------------
DIFFERENCE                          0.000
-----------------------------------------
REPORTED COMMISSIONS OWED       -5115.00
CALCULATED COMMISSIONS OWED     -5115.160
-----------------------------------------
DIFFERENCE                          0.160
-----------------------------------------
REPORTED PAYROLL AMOUNT          5115.00
CALCULATED PAYROLL AMOUNT       -5115.160
-----------------------------------------
DIFFERENCE ***** PROBLEM *****  10230.160
-----------------------------------------

***** SERIOUS COMMISSION ERRORS - SEE ABOVE *****
```

2 If rule ARWARN06 is not already, open it now.

3 Place the cursor on the line in rule ARWARN06 that begins with the string EXECUTE FINISH UPON, and press **PF08/PF20**, or enter the DOWN command, to display that line at the top of the screen.

Figure 100 Rule ARWARN06 EXECUTE FINISH Block (Display 1)

```
         LIBRARY : CTBP.PROD.RULES                                RULE : ARWARN06
 COMMAND ===>                                                     SCROLL===> CRSR
 +---------------------------------- BROWSE -----------------------------------+
 | EXECUTE FINISH   UPON                                                    C   |
 | ON DATA                                                                      |
 | ALWAYS                                                                       |
 |   DO SET      = I=1                                                      C   |
 |   DO PRINT    = TOTAL # OF SALESPEOPLE=%%TOTPEOPLE                   F   C   |
 |   DO PRINT    = %%BLANK                                              F   C   |
 | LABEL: TEST                                                                  |
 | IF       I GT TOTPEOPLE                                                  C   |
 |   DO GOTO     LABEL ENDJOB                                                   |
 | ELSE                                                                         |
 |   DO PRINT    = **** COMMISSION CONTROL - %%NAME%%NUMBER%%I ****     F   C   |
 |   DO PRINT    = REPORTED SALES               %%MTHSLS%%NUMBER%%I     F   C   |
 |   DO PRINT    = CALCULATED SALES                %%SUM1%%NUMBER%%I    F   C   |
 |   DO PRINT    = ---------------------------------------------        F   C   |
 |   DO SET      = DIFF1=ABS(SUM1%%NUMBER%%I-MTHSLS%%NUMBER%%I)             C   |
 |   DO PRINT    = DIFFERENCE                           %%DIFF1         F   C   |
 |   DO PRINT    = ---------------------------------------------        F   C   |
 |   DO SET      = COMMCALC=YTDSLS%%NUMBER%%I*0.04                          C   |
 |   DO PRINT    = REPORTED COMMISSIONS OWED    %%YTDOWE%%NUMBER%%I     F   C   |
 |   DO PRINT    = CALCULATED COMMISSIONS OWED     %%COMMCALC           F   C   |
 PLEASE FILL IN RULE DEFINITION.                                          10.38.50
```

The code shown in this screen produced the following lines of the report:

```
TOTAL # OF SALESPEOPLE=3

**** COMMISSION CONTROL -  HARVEY J. JONES  ****
```

4 Place the cursor on the line in rule ARWARN06 that begins with the string DO SET = COMMCALC=YTDSLS%%NUMBER%%I*0.04 and press **PF08/PF20**, or enter the DOWN command, to display that line at the top of the screen.

Figure 101 Rule ARWARN06 EXECUTE FINISH Block (Display 2)

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN06
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
|   DO SET      = COMMCALC=YTDSLS%%NUMBER%%I*0.04                          C   |
|   DO PRINT    = REPORTED COMMISSIONS OWED    %%YTDOWE%%NUMBER%%I    F   C   |
|   DO PRINT    = CALCULATED COMMISSIONS OWED     %%COMMCALC          F   C   |
|   DO SET      = DIFF2=ABS(COMMCALC-YTDOWE%%NUMBER%%I)                    C   |
|   DO PRINT    = -------------------------------------------        F   C   |
|   DO PRINT    = DIFFERENCE                          %%DIFF2          F   C   |
|   DO PRINT    = -------------------------------------------        F   C   |
|   DO SET      = PAYCALC=COMMCALC-YTDCOM%%NUMBER%%I                       C   |
|   DO PRINT    = REPORTED PAYROLL AMOUNT      %%PAYAMT%%NUMBER%%I    F   C   |
|   DO PRINT    = CALCULATED PAYROLL AMOUNT       %%PAYCALC           F   C   |
|   DO SET      = DIFF3=ABS(PAYCALC-PAYAMT%%NUMBER%%I)                     C   |
|   DO PRINT    = -------------------------------------------        F   C   |
| IF       DIFF3 = 0                                                       C   |
|   DO PRINT    = DIFFERENCE                          %%DIFF3          F   C   |
| ELSE                                                                         |
|   DO PRINT    = DIFFERENCE ***** PROBLEM *****  %%DIFF3              F   C   |
|   DO PRINT    = -------------------------------------------        F   C   |
|   DO PRINT    = %%BLANK                                              F   C   |
| IF       DIFF1 > 100 OR DIFF2 > 100 OR DIFF3 > 100                       C   |
|   DO SET      = ERRORCOUNT=ERRORCOUNT+1                                  C   |
PLEASE FILL IN RULE DEFINITION.                                        17.36.48
```

The code shown in this screen produced the following lines of the report:

```
REPORTED COMMISSIONS OWED       -5115.00
CALCULATED COMMISSIONS OWED     -5115.160
-------------------------------------------
DIFFERENCE                          0.160
-------------------------------------------
REPORTED PAYROLL AMOUNT          5115.00
CALCULATED PAYROLL AMOUNT       -5115.160
-------------------------------------------
DIFFERENCE ***** PROBLEM *****  10230.160
```

5 Place the cursor on the line in rule ARWARN06 that begins with the string `IF DIFF1 > 100 OR DIFF2 > 100 OR DIFF3 > 100` and press **PF08/PF20**, or enter the DOWN command, to display that line at the top of the screen.

Figure 102 Rule ARWARN06 EXECUTE FINISH Block (Display 3)

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN06
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| IF       DIFF1 > 100 OR DIFF2 > 100 OR DIFF3 > 100                       C   |
|   DO SET      = ERRORCOUNT=ERRORCOUNT+1                                  C   |
|   DO PRINT    = ***** SERIOUS COMMISSION ERRORS - SEE ABOVE *****   F   C   |
|   DO PRINT    = %%BLANK                                            F   C   |
|   DO PRINT    = %%BLANK                                            F   C   |
| ELSE                                                                       |
|   DO PRINT    = %%BLANK                                            F   C   |
| ALWAYS                                                                     |
|   DO SET      = I=INT(I+1)                                               C   |
|   DO GOTO     LABEL TEST                                                   |
| LABEL: ENDJOB                                                              |
| IF       ERRORCOUNT>0                                                    C   |
|   DO PRINT    = %%BLANK                                            F   C   |
|   DO PRINT    = %%BLANK                                            F   C   |
|   DO PRINT    = ***** SERIOUS COMMISSION ERRORS *****              F   C   |
|   DO PRINT    = ***** TOTAL NUMBER OF ERRORS %%ERRORCOUNT *****    F   C   |
|   DO TERMINAT = NOTOK    COD 0999                                          |
| ELSE                                                                       |
|   DO PRINT    = %%BLANK                                            F   C   |
|   DO PRINT    = %%BLANK                                            F   C   |
PLEASE FILL IN RULE DEFINITION.                                      17.41.15
```

The code shown in this screen produced the following line of the report:

***** SERIOUS COMMISSION ERRORS - SEE ABOVE *****

6 After analyzing and printing the balancing report for one salesperson, the counter, I, is incremented by 1. This variable is used to signify the salesperson number. Processing then goes back to the label TEST, where the rule checks if it has processed all of the data for all salespeople. If yes, processing continues with ENDJOB.

7 Place the cursor on the line in rule ARWARN06 that begins with the string LABEL: ENDJOB and press **PF08/PF20**, or enter the DOWN command, to display that line at the top of the screen.

Figure 103 Rule ARWARN06 LABEL: ENDJOB

```
        LIBRARY : CTB.PROD.RULES                                RULE : ARWARN06
COMMAND ===>                                                    SCROLL===> CRSR
+---------------------------------- BROWSE -----------------------------------+
| LABEL: ENDJOB                                                               |
| IF       ERRORCOUNT>0                                                   C   |
|   DO PRINT    = %%BLANK                                             F   C   |
|   DO PRINT    = %%BLANK                                             F   C   |
|   DO PRINT    = ***** SERIOUS COMMISSION ERRORS *****               F   C   |
|   DO PRINT    = ***** TOTAL NUMBER OF ERRORS %%ERRORCOUNT *****     F   C   |
|   DO TERMINAT = NOTOK    COD 0999                                           |
| ELSE                                                                        |
|   DO PRINT    = %%BLANK                                             F   C   |
|   DO PRINT    = %%BLANK                                             F   C   |
|   DO TERMINAT = OK       COD 0000                                           |
| ========================================================================== |
| EXECUTE CHKNAME  UPON                                                   C   |
| ON DATA                                                                     |
| IF       SALESNO NE LASTSALES                                           C   |
|   DO SET      = TOTPEOPLE=INT(TOTPEOPLE+1)                              C   |
|   DO SET      = NUMBER%%TOTPEOPLE=SALESNO                               C   |
|   DO SET      = LASTSALES=SALESNO                                       C   |
| ========================================================================== |
| EXECUTE          UPON                                                   C   |
PLEASE FILL IN RULE DEFINITION.                                        17.43.56
```

The code shown in this screen produced the following lines of the report and terminates the job, depending on the results.

Figure 104 Sample Output Produced by LABEL: ENDJOB

```
**** COMMISSION CONTROL -  ALI MCGRATH      ****
REPORTED SALES                  15000.00
CALCULATED SALES                15000.000
-------------------------------------------
DIFFERENCE                          0,000
-------------------------------------------
REPORTED COMMISSIONS OWED       13515.16
CALCULATED COMMISSIONS OWED     13515.160
-------------------------------------------
DIFFERENCE                          0,000
-------------------------------------------
REPORTED PAYROLL AMOUNT         11632.03
CALCULATED PAYROLL AMOUNT       11632.030
-------------------------------------------
DIFFERENCE                          0,000
-------------------------------------------

**** COMMISSION CONTROL -  MARYJANE INKSTER ****
REPORTED SALES                  16500.00
CALCULATED SALES                16500.000
-------------------------------------------
DIFFERENCE                          0,000
-------------------------------------------
REPORTED COMMISSIONS OWED       39995.16
CALCULATED COMMISSIONS OWED     39995.160
-------------------------------------------
DIFFERENCE                          0,000
-------------------------------------------
REPORTED PAYROLL AMOUNT         39995.16
CALCULATED PAYROLL AMOUNT       39995.160
-------------------------------------------
DIFFERENCE                          0,000
-------------------------------------------

***** SERIOUS COMMISSION ERRORS *****
***** TOTAL NUMBER OF ERRORS 1.000 *****
```

CONTROL-M/Analyzer Capabilities Summary

In defining actual rules to handle the realistic problems presented in this chapter, you used ON blocks and DO parameters. These CONTROL-M / Analyzer capabilities are summarized below:

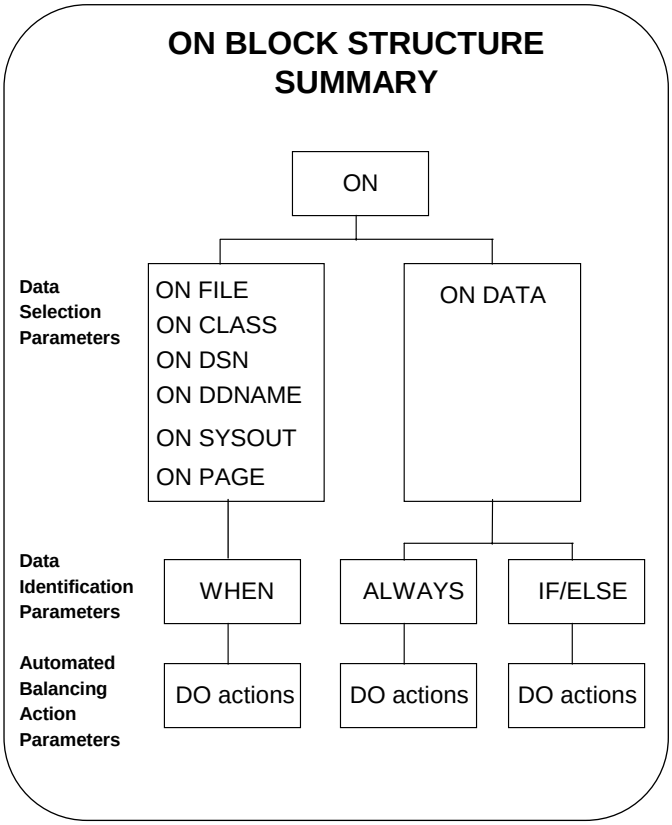
ON Blocks

You've seen that every rule is made up of ON blocks. There are two main types of ON blocks:

- blocks that process an external data source, extracting information, and
- blocks that work on internal data, such as variables

As Figure 105 illustrates, there are several types of external data sources:

Figure 105 ON Block Structure Summary



In Figure 106, the two types of data sources (ON SYSOUT and ON FILE) that were used in solving the six problems described in this chapter are checked.

While the other three types of data sources (ON CLASS, ON DDNAME, and ON DSN) would require different ON specifications, the rule definitions that would use these data sources would be nearly identical to those using ON SYSOUT and ON FILE.

Figure 106 Data Sources That ON Statements Can Process

ON STATEMENTS

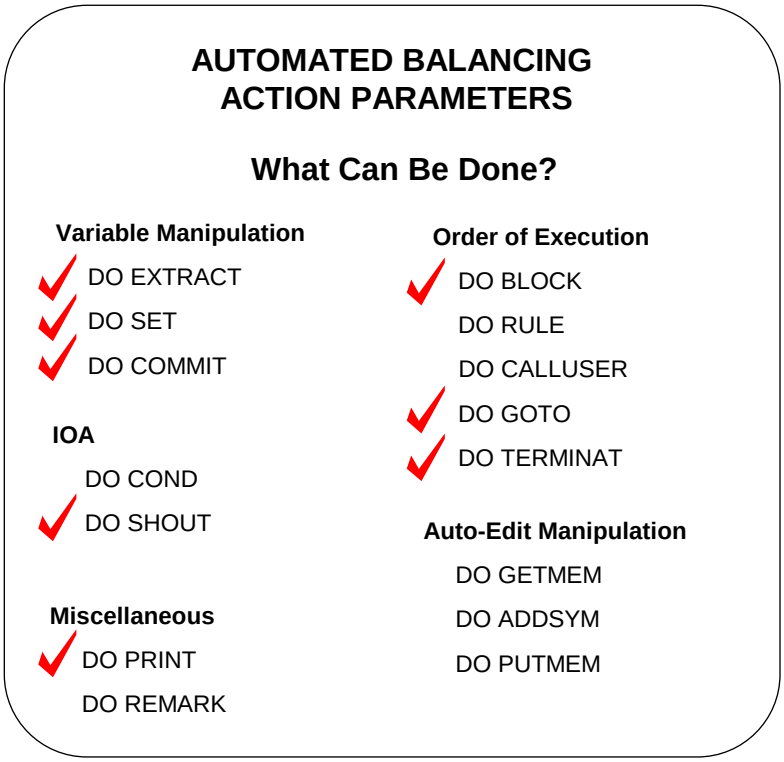
Which data sources can be processed?

✓ ON SYSOUT	Outputs of a DD statement e.g., Reports
✓ ON FILE	Sequential Data Set
ON CLASS	Outputs of job belonging to a class of SYSOUTS
ON DDNAME	Sequential Data Set specified in JCL DD statement of the current step
ON DSN	Compressed SYSOUT dataset created by CONTROL-D

DO Parameters

Figure 107 illustrates the DO parameters that can be used in CONTROL-M / Analyzer balancing rules. The DO parameters that were used in solving the six problems in this chapter are checked.

Figure 107 What Automated Balancing Action Parameters (DO Statements) Can Do



From the diagram, you see that by performing the exercises in this guide, you have

- performed a great deal of variable manipulation, using DO EXTRACT, DO SET, and DO COMMIT

- used DO SHOUT to invoke the IOA SHOUT facility

  The DO COND adds or deletes conditions from the IOA Conditions file.

- printed reports, using DO PRINT

  DO REMARK creates a one line remark which is placed in the Invocation Report and the Rule Activity display.

- modified the order of execution of the rule, using DO BLOCK, DO GOTO, and DO TERMINAT

  While you have not called other rules (DO RULE), or user-defined routines within a rule (DO CALLUSER), these action parameters are very similar to the others.

- You did not need to use “AutoEdit” manipulation actions (DO GETMEM, DO ADDSYM, DO PUTMEM) in this demo. These special purpose variables are explained in the *CONTROL-M/Analyzer for z/OS User Guide.*

Chapter 3

Preparing for Production — Application

This chapter includes the following topics:

Overview 150
For Newcomers to INCONTROL Products and the IOA Environment 150
Integrated IOA Activity Log 151
Integrated Management Reporting. 151
INCONTROL Products. 152
Suggested Conventions 153
Finding Data in Reports and Files 157
Saving Data as Opposed to Using Local Variables 159
Implementation Hints 161
Rule Definition 161
Quality Control and Avoiding Errors 164
Invoking CONTROL-M / Analyzer from Other Applications in the Processing Environment. 166
Invoking the Rule from a Job 166
Invoking the Rule from a Balancing Mission 166
Structured Approach for Implementation 167
Rule Worksheet A 168
Rule Worksheet B 169
Mission Worksheet 170
Database Variable Worksheet 171

Overview

The purpose of this chapter is to provide suggestions and recommendations for implementation of CONTROL-M / Analyzer applications in your production environment. Its primary audience is those responsible for defining balancing applications and rules, and setting standards for user application development.

If you require further detail on any of the options discussed in this chapter, refer to the *CONTROL-M/Analyzer for z/OS User Guide*.

The first section of this chapter is for those who are not familiar with INCONTROL products or the IOA environment. If you are already familiar with INCONTROL and IOA, you may want to continue with “Finding Data in Reports and Files” on page 3-157.

For Newcomers to INCONTROL Products and the IOA Environment

The Integrated Operations Architecture (IOA) is the software architecture of the INCONTROL family of Automated Systems Operations products. The purpose of IOA is to establish an operations software environment in which the goal of true unattended operations can be realized.

You interact with all INCONTROL products through the use of a single user-friendly online interface. The interface gives you the capability to control and administer all functions in the automated data center environment from a single terminal. This single user interface is available in every online environment, including TSO / native, TSO / ISPF, CA-ROSCOE, CICS, IMS / DC, COM-PLETE, and VTAM.

All INCONTROL products use a single integrated database containing all production information relating to the operation of all data center functions. This single database facilitates the automation of the links, and the scheduling of events, among the INCONTROL products.

All INCONTROL products are centrally scheduled and coordinated by a single scheduling facility. In this manner, you can specify the entire process through a single scheduling screen. For example, through that single scheduling screen you can

- schedule a production job
- conduct balancing and quality control of the output report for a job
- restart a job in case of job failure
- direct the printing and distribution of reports

Further, the single scheduling facility enables the specification of dependent relationships across functions. That is, the occurrence of an event in one automated function, such as the printing of a report, can be made dependent on the occurrence of an event in another function, such as the successful finish of a job.

INCONTROL products are distributed on one installation tape, with an accompanying unified installation process. An automatic maintenance procedure keeps all INCONTROL products at the same maintenance level.

Integrated IOA Activity Log

The integrated IOA Activity Log contains automatically generated messages that record every significant event in the activities supervised by different INCONTROL products. The IOA Log can be viewed online.

Integrated Management Reporting

IOA provides a special report generation language called the KeyStroke Reporting Language (KSL). KSL can be used to generate cross-product reports from a single database.

INCONTROL Products

The integration of INCONTROL products offers you the benefits of

- a single application
- a common user interface
- a single database

The INCONTROL family of products includes:

Table 6 List of INCONTROL Products (part 1 of 2)

Product	Description
CONTROL-M	**Automated Production Control and Scheduling System** Manages and automates the setup, scheduling and execution of jobs in the data center.
CONTROL-M/Restart	**Restart Management System** Automates the activities that must be performed when restarting failed jobs, including the scratching and uncataloging of datasets created by failed jobs.
CONTROL-M/Tape	**Removable Media Management System** Increases utilization of removable media and controls retention periods. Prevents misuse of media, and provides tape library and vault control.
CONTROL-M/Analyzer	**Automated Information Integrity System** Performs in-stream validation, accuracy, and reasonability checks on information used by data center production tasks (for example, reports, databases).
CONTROL-D	**Output Management System** Automatically schedules and controls every aspect of report processing and distribution, including report decollating, bundling, printing, online viewing, and archiving.
CONTROL-V	**Quick Access Archive Viewing System** Provides online access to archived reports and documents by indexed data retrieval.
CONTROL-D/ Page On Demand	**Report Retrieval and Display System** Enables end users to retrieve and view pages of reports that reside on mainframe storage in real time. Indexed reports can be retrieved by index name and value. AFP and XEROX reports can also be retrieved and displayed using CONTROL-D / WebAccess Server or CONTROL-D / Page On Demand API.

Table 6 List of INCONTROL Products (part 2 of 2)

Product	Description
CONTROL-D/Image	**Image Output Management System** Enables output from commercial imaging equipment to be imported into either CONTROL-D or CONTROL-V for decollation, distribution and viewing, and into CONTROL-V for archiving and indexed retrieval.
CONTROL-O	**Console Automation System and Desired State Monitoring System** Monitors and automatically responds to messages, commands, and dataset events, as well as various other system events. The CONTROL-O / COSMOS feature allows for status monitoring while maintaining all critical system objects in a desired and ideal status.

Suggested Conventions

You may find it advantageous to have a method of organizing groups, variables, and libraries. If you organize in this way before your applications begin to grow, it will be easier to find and update information within CONTROL-M / Analyzer.

Naming Groups

The ability to organize your work into groups is an important facility of INCONTROL products.

The group concept is discussed in detail in Chapter 1, and in the *CONTROL-M/Analyzer for z/OS User Guide*.

The naming convention for groups should

- mirror your organization
- reflect how CONTROL-M / Analyzer applications are defined and administered
- consider the database and systems standards of your organization

Group names can be up to 20 characters.

You may also use an underscore character within the group name, to make the name easier to read and understand.

Shown below are examples of the types of group names you may want to devise:

GROUPS defined by functional responsibility in the work place

- MIS: Data processing
- ACCOUNTING: Accounting department
- INVENTORY: Inventory management
- SALES: Sales operations
- EXECUTIVE: Directors, Vice Presidents, President
- MANUFACTURING: Production
- PURCHASING: Purchasing
- ADMIN: Administration

GROUPS defined by MIS application area

- INVOICING: Invoicing
- ACCT_REC: Accounts Receivable (collections)
- ACCT_PAY: Accounts Payable (vendor payments)
- GNRL_ACCT: General accounting applications
- PRODCTL: Production control
- CAD: Computer-aided design

GROUPS defined by database organization

- CUSTOMER: Clients
- VENDOR: Suppliers
- PRODUCTS: Products / Services of the company
- EMPLOYEES: Employees
- ACCOUNTS: Accounting chart of accounts
- PARTS: Items used in production

Naming Variables

Variables are automatically part of the group in which they were created. Their names should therefore reflect their function within the group. Variable names can be up to 20 characters.

Because you can potentially create a very large number of variable names, it is important that you using naming conventions that can help you readily identify a variable. The following are some general suggestions for naming variables:

- Check the standards of your organization.

  There may already be standard methods of naming items in your systems department.

- Use as descriptive a name as possible, but at the same time use the smallest number of characters.

  Defining rules can become long and tedious if you must continuously enter variable names that are 20 characters long. On the other hand, if all your variable names were only 2 characters long and had no readily apparent meaning, it would be much harder to read, understand, and modify the variable names within a rule.

- Use names that are meaningful to everyone who is using the same application within the CONTROL-M / Analyzer environment, such as those who are defining rules, operations staff, systems staff, and management.

- Be consistent.

- Remember that the group concept provides a way of keeping related variables together. There is no need to duplicate a group name, such as division code, within a variable name.

Here are some examples:

VARIABLES, With a Prefix or Suffix of a Division or Department Code

Assume that you are defining a balancing application, and want to store data in the CONTROL-M / Analyzer database, and segregate that data by division. You will also typically retrieve, review, and update the data by division. This is similar to how the rules in Chapter 2 are defined.

There are two possible naming conventions:

- Prefix each database variable name with the division code.
- Add the division code as a suffix to the variable name.

Some examples are:

```
D001_COST                COST_001
D002_COST                COST_002
D003_COST      or        COST_003
D001_SALES               SALES_001
D002_SALES               SALES_002
D003_SALES               SALES_003
```

The issue of which convention is better can be determined when you answer the following questions:

- How do you usually retrieve and view these data?
- Do you retrieve the data by division, or by the type of variable, such as cost, or sales?

In the first case, the division code as a prefix is better. In the latter case, the suffix approach is more helpful.

VARIABLES, Associated with an "Owner"

Many organizations attach each variable to an "owner," meaning either the department or job title responsible for maintaining the data. Some examples are:

```
PURCH_SUPPLIES
PURCH_CAPITAL
PERSNL_SALARIES
PERSNL_BENEFITS
```

In these examples, the two departments, PURCH (Purchasing) and PERSNL (Personnel), each "own" their respective variables. This naming convention may be better handled by creating separate groups for the purchasing and personnel departments, and eliminating the department prefix from the variable name.

While naming convention possibilities are extremely flexible, it is important to adopt them from the beginning. It is much easier to define variables once, and not have to redefine them later.

Organizing Libraries

Within CONTROL-M/Analyzer, two main libraries are used and referenced:

- RULES: Where the rules which you have defined are kept.
- BALMIS: Where missions are defined.

While you could theoretically put all the above information in one library, there are better ways of organizing this information. Following are some suggestions for organization methods.

Organize Libraries by Group or Application

This is a logical way of administering security, tracking online storage usage, and providing MIS services.

This means that you could have, for example, a rule library for each group or application, a mission library for each group, and JCL libraries for each group. Users who want to view information about their application know exactly where to look, and have security privileges on their libraries. On the other hand, unauthorized users are prevented from accessing and changing rules or missions outside their own applications.

Keep Separate Libraries for Production and Testing

Production libraries should permit very limited access, and should be free of non-working or out-of-date items. They should only contain the items relevant to doing actual balancing functions.

Test libraries tend to have wide access, and also tend to quickly fill up with items that need frequent cleanup, such as rules you never implemented, one-time functions, rules, and JCL used for training.

Finding Data in Reports and Files

Usually, when performing a balancing job, you are given a report or file to work with, and that report or file clearly was not designed with CONTROL-M / Analyzer in mind. You may be working with a report that was designed a long time ago, or one that is very complex.

Some of the challenges you may encounter, when searching for data in a report or file are:

- knowing when and how to use the NOT EQUAL parameter
- the data for which you are searching do not always appear in the same line or place on a report page
- difficulty in uniquely identifying data in a WHEN clause
- knowing how to use the TYPE AV and TYPE SU parameters

  These parameters operate only on numeric data. Therefore, variables that are extracted from a file or report with one of these operators must be numeric.

The following are some suggestions for alternate approaches:

Using the NOT EQUAL Condition to Identify Data

Sometimes a group of report lines or file records can be readily identified because they are the only lines that do not contain certain characters. See, for example, Problem 4 in Chapter 2.

In this example, two types of file records exist—detail records and total records. One way of separating the two types of records is to describe detail records as not having the characters “TOTALINVOICES” in columns 3 through 15.

In using a NOT EQUAL condition, it is important to be 100% precise when specifying the column width. You cannot specify a column range where the string can be found. Instead, you must specify the exact columns where the string is located. This is because CONTROL-M / Analyzer compares the string which it has extracted from the file, character by character, to the string that you specified in the WHEN statement.

For example, what would happen if you were to specify a WHEN parameter to search for detail records as described above? Assume that your WHEN statement specifies columns 1 through 20 for a string NOT EQUAL to TOTALINVOICES. All the detail lines meet the NOTEQUAL search criteria, which is exactly what you expected. But, when the TOTALINVOICES line in the report is examined, it also meets the criteria. This is because a blank in column 1 had been compared to the character “T” from the WHEN statement. Because the blank and the character “T” are not equal, the total line is therefore presented as a detail line.

Data in Different Locations in a Report

Sometimes a line of data that you are looking for cannot be uniquely identified by itself. For example, you might have a sales analysis report that shows detailed sales information for a product, followed by a line of total amounts for that product. Further, the total line has no special characters, such as “TOTAL SALES,” with which it can be identified. Also, the number of detail lines occurring before the total line is printed may be any number. Therefore the total line can occur almost anywhere on the page.

For CONTROL-M / Analyzer to be able to find the line, you must establish a unique relationship between this total line and another line.

For example, perhaps the line can always be found two lines before the next product heading, which contains the characters SALES ANALYSIS FOR PRODUCT.

You could then define a WHEN clause that searches for the string “SALES ANALYSIS FOR PRODUCT,” and then define a DO EXTRACT from that line, minus 2.

Data that Must Be Associated with External Data before Balancing

Assume, for example, that you have a summary report of all of the accounting transactions for your organization for three months. The report shows the three month time period range as a heading line on the report, but the exact date is not shown within each detail line. However, each detail line does have a transaction number from which it is possible to determine a transaction date by looking up this number in a database, such as DB2, ADABAS, or ORACLE.

The auditing department has asked for a balancing report showing all the transactions for one account number, for one month.

Assume that you can define a WHEN parameter to select all transactions for a given account number from the report. However, to determine the time period to which a transaction belongs, you need to

- examine the detail line, which contains a transaction number
- find the date with which this transaction is associated

  To do this, you have two alternative approaches:

  — You can set up a table of variables that provides a transaction date for a range of transaction numbers, and look up the transaction number in this table.

  — You can issue a DO CALLUSER parameter. This would call a user-written routine that would look up a transaction number in a database, and return a date to your rule.

- use groups of IF statements to determine if this transaction meets your criteria

Problem 6 in chapter 2 illustrates an example of how to set up and use a table of variables.

Saving Data as Opposed to Using Local Variables

Local variables are those variables that are used within a rule, but are not saved permanently. At the time your rule terminates, whatever values that existed in its local variables are wiped from memory.

You may instead want to save these variables in the CONTROL-M / Analyzer database, or in an AutoEdit PDS library member, for the following reasons:

- to reference it within other rules
- to retain it for an audit trail or for other historic purposes
- to have the information online, so you can retrieve at any time
- to reference it by other INCONTROL products
- to use for “debugging,” meaning, to reconstruct what actually happened when a rule has been executed

The following discusses the advantages and disadvantages of either saving variables to the CONTROL-M / Analyzer database, or alternatively, an AutoEdit PDS library member. The nature of the data you need to save in a balancing application will help you determine which of these methods to use.

Saving Data in the CONTROL-M/Analyzer Database

When deciding if you should store variables in the CONTROL-M / Analyzer database, consider the following questions:

- Will the data be used by another rule or application?
- Is the data required for online viewing?
- Are you reasonably certain that the data will be required at some time in the near future?

If you cannot affirmatively answer at least one of these questions, you should consider alternatives to storing data permanently in the database.

An alternative to storing those variables that are only required for debugging or audit trail purposes, is to print the values of the variables in a report. There are two standard reports provided when a rule executes—the SYSUSER report and the SYSPRINT Invocation Report. For debugging and audit trail purposes, it is probably better to print these local variable values using DO PRINT parameters to the Invocation Report.

Saving Data in AutoEdit Members

There are several advantages and disadvantages to saving data in an AutoEdit member as opposed to the CONTROL-M / Analyzer database.

The advantages are:

- AutoEdit members are available to other INCONTROL products. They are used mainly by CONTROL-M for JCL setup.
- AutoEdit members can contain more than one variable. Therefore, a group of variables can be retrieved into memory by one DO GETMEM parameter.
- New variables can be added dynamically to an AutoEdit member by the DO ADDSYM parameter.
- The values in the member can be edited and corrected.

The disadvantages are:

- Multiple generations, with audit trail information, cannot be automatically maintained by CONTROL-M / Analyzer.
- No automatic rollback mechanism is available for AutoEdit members.

Implementation Hints

Rule Definition

When you are ready to define your first CONTROL-M / Analyzer rules, consider the following implementation hints:

Build and test your rule piece by piece

- Start with simple WHEN statements that extract data from a file or report. Test the success of the WHEN statements by printing some data you've extracted. The Invocation Report will tell you if you have successfully defined the ON block—meaning, if the WHEN search criteria were actually met.
- Add a simple DO BLOCK parameter to analyze the data that was found in the WHEN parameter. Use DO PRINT parameters to verify that each DO action you are taking has done what you expected it to do. Test your rule again.
- Add an ON block that will handle the termination of the rule. Test the rule again, and check the Invocation Report to see if the rule ended as you anticipated, with the proper user code.
- Review the many examples of rule definition in this guide and in the *CONTROL-M/Analyzer for z/OS User Guide*.

Do research before working online

Before beginning to work online, determine the following:

- what external data you need to analyze, such as reports, files, or CONTROL-D compressed files
- what will be the file and report layout
- how do you identify the data you want to analyze
- what internal data, such as a database and AutoEdit variables, do you need
- what analysis do you want to perform
- what database or AutoEdit variables do you need to commit if the rule executes OK

- under what conditions do you consider the balancing result OK, tolerable, or NOTOK—and what codes will be used to indicate the various possible results?

Many people prefer to work out these details on paper, before approaching the computer terminal. To assist you, a series of organized forms has been provided in the section titled “Structured Approach for Implementation” on page 3-167.

Analyze unexpected results

When the unexpected happens after executing a rule, use the following facilities to trace the steps the rule took:

- **Run the TRACE Facility**

  Run the TRACE facility to examine CONTROL-M / Analyzer commands while a rule is executing and after its termination. When a trace run is requested, and during rule execution, the CONTROL-M / Analyzer commands of the rule being executed are printed by block. The TRACE facility is also discussed in detail in the *CONTROL-M/Analyzer for z/OS User Guide*.

- **Use the Invocation Report**

  The Invocation Report tells you which ON blocks have been executed, from the very start of the rule until the rule terminated. This is done using an ENTERING BLOCK print line, which shows the name of the ON block executed, and what type of block it is, such as FILE, SYSOUT, or DATA. The report also shows each line printed with a DO PRINT parameter, each variable committed with a DO COMMIT parameter, and other selected actions. At the end of the rule, the Invocation Report shows how the rule ended, how many variables were committed, and any resulting user code.

- **Add parameters to the rule to print values of local variables and other information that can help you analyze what the rule is doing**

  There are some actions, such as results of DO SET parameters, which the Invocation Report does not automatically show. If you believe that your unexpected results are because of changes in local variable values caused by a DO SET parameter, you can easily trace this. All that you need to do is add selective DO PRINT instructions in your rule to print the values of these local variables as they are changed.

  If you are using more than one such DO PRINT lines then you may wish to identify where they came from in the printout by putting a line number in the remark, or putting the actual DO SET instruction in the remark.

- **Change a parameter specification slightly**

For example, you may be using WHEN clauses to search for a string in your report. Unexpectedly, CONTROL-M/Analyzer does not find the data for which you are searching. Yet when you look at the report, you see that the data exists, and the characters for which you are searching are exactly as you expected them to be.

By making a small change in your parameter specification, it may give you a clue as to why the desired data are not being located. For example, you may have defined the length of the string, or the positions (columns, or line numbers to search in), incorrectly. Try specifying a broader or a narrower column or line range.

Try changing a .NE. (not equal) search string to an .EQ. (equal) or a .GE. (greater than or equal to) condition. Try embedding the string you are searching for in quotes if it contains any blanks or special characters.

- **Check each WHEN statement separately**

  If you have a complex compound WHEN statement, try checking each WHEN specification separately. First, ensure that the outermost WHEN finds its string in each line of the report or file. When you are sure that this specification is working, add the next WHEN statement.

- **Ask another CONTROL-M/Analyzer user for help**

  As the CONTROL-M/Analyzer expertise builds in your organization, help will be readily at hand. You may also take advantage of support from your supplier.

Use short ON blocks rather than long ON blocks

There are times when you can write an entire rule within one or two ON blocks. There are two occasions when it may be desirable to split one long ON block into two or more ON blocks:

- **Unnecessary duplication within the rule definition**

  Assume the following:

  — every time the rule prints a line, it follows the printed line with two blank lines and two lines consisting of all dashes

  — the rule prints 30 different lines in the report

  With all of the DO PRINT instructions in one ON block, there would be an extra 120 DO PRINT instructions (4 times 30) for the separator lines. On the other hand, if you set up a separate PRINT block in the rule, you would have to write the print instructions for the separator lines only once and then issue a DO BLOCK instruction 30 times, saving a net of 90 instructions.

- **Better tracing through the Invocation Report**

  The Invocation Report shows which ON blocks of your rule have been executed. Sometimes an ON block uses labels so that a separate part of the ON block can be executed repeatedly. See Problems 4, 5, and 6, in Chapter 2 for examples of this structure.

  In this case, the processing flow that occurs when these labelled parts of an ON block are executed would not be readily discernible from the Invocation Report. If, instead, you split the ON block into multiple ON blocks, it may be easier to use the report to trace what is happening in the rule.

Using the CONTROL-M/Analyzer Rule Definition Editor versus the System Editor

Rules can be defined using the CONTROL-M / Analyzer Rule Definition facility or by using a standard system editor. Until you are very familiar with the syntax of CONTROL-M / Analyzer rules, it is highly recommended that you use the CONTROL-M / Analyzer Rule Definition facility.

The CONTROL-M / Analyzer Rule Definition facility contains many helpful messages. When you omit a parameter, or fill it in incorrectly, the rule definition editor informs you immediately. It also uses a fill-in-the-blanks approach, avoiding the need to know the syntax of the parameters.

Quality Control and Avoiding Errors

CONTROL-M / Analyzer is a product designed to enhance the data balancing and quality control function within an organization. As such, it should not be exempt from its own quality control procedures.

Within a rule, consider using the quality control techniques described in this section. These techniques can be used during testing of the rule and then removed after testing. For future maintenance, however, you might also consider leaving these quality control techniques in place after testing, but modifying them temporarily so that none of their output is written to the Invocation Report.

Use totals to check that your data selections (WHEN parameters) and other DO actions are correct

You can total numeric fields within the report or file being analyzed. These total amounts can be used to check that an input file or report contains all of the data that you are expecting. This is also a way of verifying that you are, in fact, processing the correct file from the correct source and date.

Once you have established some initial totals, even on fields that you may not be verifying, you can use these as a base or control figure to compare against other balancing actions in the rule. For example, if you are extracting certain sales figures that you believe represent about 10% of the total sales in a report, you can calculate a total on the report lines extracted to see if they do represent about 10% of the total dollar amount of sales.

Have another person check any output which you are producing

When you are focusing on developing rules, you may occasionally miss some obvious errors. An important rule of quality control is to always have one person check the work of another person.

When checking the validity of a rule, use test reports or files which contain many test cases

The greater the variety of error conditions in your test report or file, the greater the chances that your rule will work as expected on the live report. On the other hand, if your test report or file is too large, you will not have the time to check everything it is doing.

After testing with your own "made-up" report or file, use a live report or file

No matter how many conditions you may think of for testing your rule, there are usually some surprises waiting for you in the real data. Before implementing your rule in production, it is a good idea to test it on some actual reports or files, and review the results with users. You may use the CONTROL-M / Analyzer "simulation" mode for this purpose.

Keep the test reports and files, and re-use them when any changes are made to a rule. Compare output after the rule change to previous output

To ensure that you have not overlooked anything when you change a rule, you should have a "base of comparison." The results of the rule before the change should be kept on file, with sample reports and files, and sample outputs produced, including the Invocation Report.

When you make a change, all outputs should be compared.

Run the TRACE facility to debug your rule

The TRACE facility produces a complete printed output of every command executing during rule processing.

Invoking CONTROL-M/Analyzer from Other Applications in the Processing Environment

Within the overall processing environment in your Data Center, you will eventually want to establish a close connection between a CONTROL-M / Analyzer balancing job and other jobs or job steps from other applications and environments.

There are many ways to invoke the CONTROL-M / Analyzer environment from other environments, which can be done

- from a job step
- from an application program
- from the CONTROL-M Production Control System
- from the CONTROL-D Output Management System
- by a CONTROL-M / Analyzer balancing mission

Invoking the Rule from a Job

Two examples of invoking CONTROL-M / Analyzer include

- adding a CONTROL-M / Analyzer step as the last step of a job, to check the outputs of that job

- executing a CONTROL-M / Analyzer rule in the middle of a job in order to check an input file prior to a database update

In these cases, you would add a job step to an already existing job in the correct location in that job. This job step would then call the CONTROL-M / Analyzer balancing rule. You would not need to define special scheduling criteria for the balancing rule, as the rule would fit into whatever scheduling criteria are specified for the existing job.

Invoking the Rule from a Balancing Mission

If your Data Center has no job scheduler application, then scheduling and other criteria can be specified within a balancing mission. In this case, a job or job step would be established to invoke the CONTROL-M / Analyzer environment, specifying the mission to be run. For more information on CONTROL-M / Analyzer balancing missions, refer to *the CONTROL-M/Analyzer for z/OS User Guide*.

Structured Approach for Implementation

Much of the effort involved in implementing a complete balancing system takes place before sitting down at a terminal to define a rule. Users are consulted, the problems are discussed, objectives are established, criteria for determining errors are documented, and desired reports and actions are designed.

To assist you before you get to a terminal, definition templates are provided here. These templates are designed to help data balancing designers organize their thoughts, and to make it subsequently easier to define rules, missions, and variables.

These templates include:

- Rule Worksheet A
- Rule Worksheet B
- Mission Worksheet
- Database Variable Worksheet

Rule Worksheet A

CONTROL-M/Analyzer RULE WORKSHEET A **Page ___ of ___**

Group	
Library	
Mission	
Category	
Rule	
Owner/User	

Rule Desc.

External Data Source

- ☐ Input File (Sequential or PDS)
  DD _____
- ☐ Input File (Sequential or PDS)
  DSN _____
- ☐ CONTROL-D Compressed Data Set
  _____
- ☐ ON Class SYSOUTS
  Class ___ ___ ___ ___
  Form ____ Ext. Wrt. ____ Dest. ____
- ☐ ON SYSOUT
  Procstep _____
  Pgmstep _____
  DD _____

Search Criteria for Data

Lines From: _____ To: _____
Columns From: _____ To: _____
Search For:

Data to Extract from Reports/Files

Name	Line	Columns	Avg/Sum

CONTROL-M/Analyzer Data Base Variables (I = Input, N = New, G = Add Generation)

Variable Name	Description	I/N/G

Termination Codes/Actions

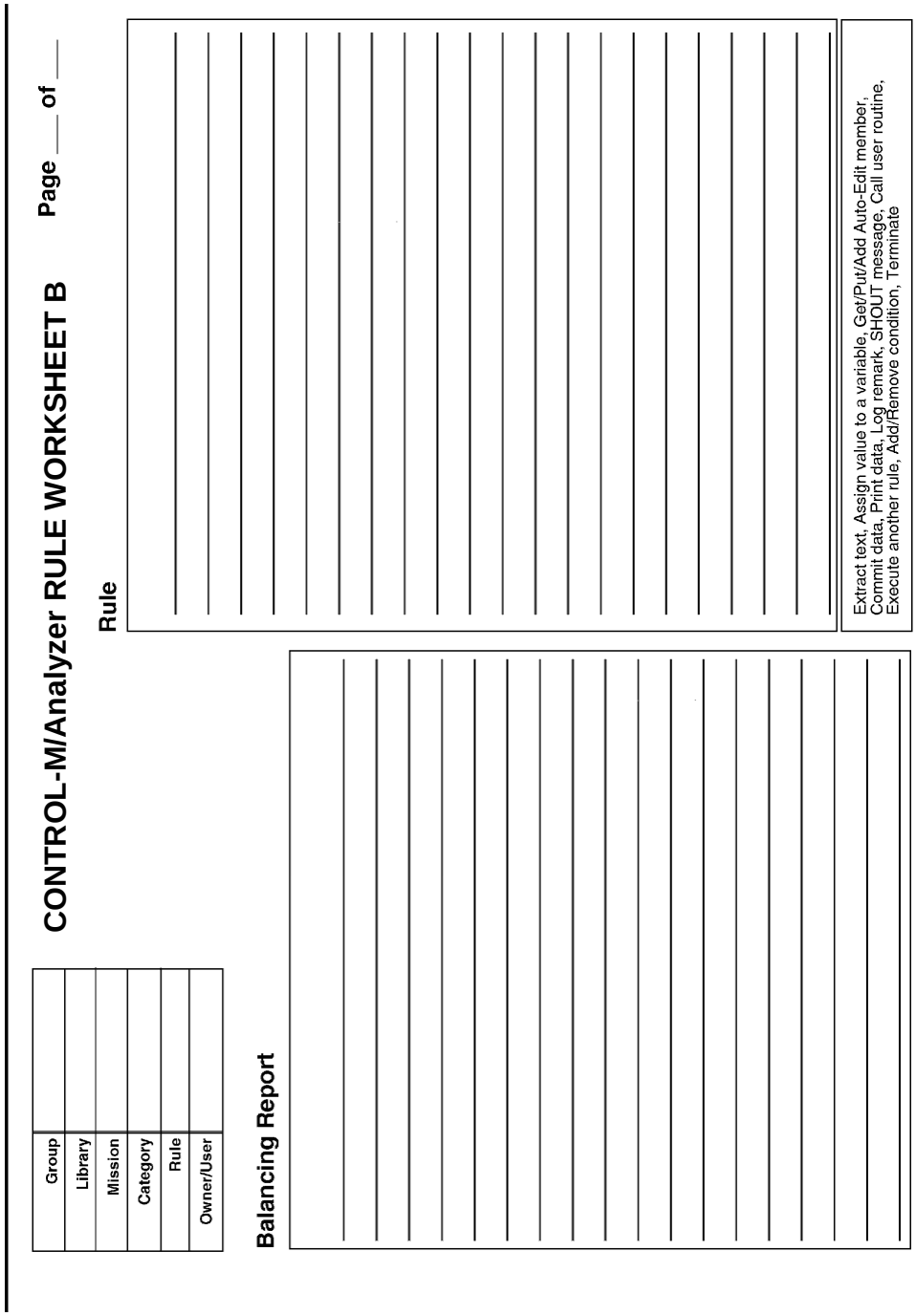
Balancing OK - Code _____

Tolerable - Code _____

NOTOK Code _____

NOTOK Code _____

Rule Worksheet B



CONTROL-M/Analyzer RULE WORKSHEET B Page ___ of ___

Group	
Library	
Mission	
Category	
Rule	
Owner/User	

Balancing Report

Rule

Extract text, Assign value to a variable, Get/Put/Add Auto-Edit member, Commit data, Print data, Log remark, SHOUT message, Call user routine, Execute another rule, Add/Remove condition, Terminate

Mission Worksheet

CONTROL-M/Analyzer MISSION WORKSHEET

Group	
Library	
Mission	
Category	
Rule	
Doc. Library	
Doc. File	
Owner/User	

Mission Desc.

_____

_____

Scope of Execution

☐ SINGLE - Once Only ☐ STEP - Any # Times/Step

☐ JOB - Any # Times/Job ☐ ALL - Any # Times

Initiated By

☐ Job/Step (Name) _____

☐ CONTROL-M

☐ CONTROL-D

☐ Program (Name) _____

☐ Other

Prerequisite Conditions

_____

_____

_____

_____

_____

Schedule

Days/Week: 1 2 3 4 5 6 7
OR All

Days/Mth: 1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31
OR All

Months: Jan Feb Mar Apr May Jun
Jul Aug Sep Oct Nov Dec
OR All

Other Dates

_____

_____

Calendar Name

_____

Time for Execution

From _____ To _____ Not Later Than _____

Retroactive Scheduling

☐ Yes ☐ No

If Yes, for Max # Days _____

Database Variable Worksheet

CONTROL-M/Analyzer DATA BASE VARIABLE WORKSHEET

Group	
Created By	
Owner/User	

Group Desc.

Variable Name	Description	Maximum Generations	Type*	Mask*

*Reserved for future use

Chapter 4

Preparing for Production—Technical

This chapter includes the following topics:

Overview 174
CONTROL-M / Analyzer and IOA Fundamentals 174
AutoEdit Members. 175
Concurrent Usage and Locking Considerations 176
Backing Up, Restoring, Restarting. 177
Use of Utilities 178
- Utility CTBJAFDL 178
- Utility IOADBF 178
- Utility CTBxxxCP 179
- Utility CTBFRM 179

Setting Up a New Day Procedure 179
- Purpose 179
- Cleanup Activities 180
- Missions 180

Overview

The purpose of this chapter is to introduce you to the software issues you might encounter when implementing CONTROL-M / Analyzer in your production environment.

This chapter is designed primarily for those responsible for installing and maintaining the CONTROL-M / Analyzer software. The issues discussed here are not relevant to any specific balancing application, but are generic to the running of CONTROL-M / Analyzer within your installation.

CONTROL-M/Analyzer and IOA Fundamentals

The *CONTROL-M/Analyzer User Guide* provides an overview of the product, and its interrelationship with other INCONTROL products. A thorough review of that information is essential for anyone who is responsible for installing or maintaining the software environment.

There are important areas of interaction between CONTROL-M / Analyzer and other automated operations products. In reviewing the *CONTROL-M/Analyzer User Guide*, the following topics should be examined in detail:

- how rules and missions are invoked
- integration with IOA
- the production environment
- the repository
- job flow
- components
- data definition

AutoEdit Members

In a computer production environment, it is often necessary to manually modify JCL code prior to the submission of a job. The JCL Setup and AutoEdit Facilities of the CONTROL-M Production Control System eliminate the need for these manual changes to the JCL. JCL Setup and the AutoEdit Facility are discussed in greater detail in the *CONTROL-M for z/OS User Guide*.

The AutoEdit facility allows you to maintain a group of variables in a PDS library member. Both the variable name and its current value are maintained. This facility allows you to:

- retrieve the AutoEdit member into memory, making its variables available to the rule
- change any of the variable values, and write all changes back to the AutoEdit member on disk
- add new variables to the AutoEdit member

For a discussion of the advantages and disadvantages of AutoEdit members, see the discussion on local variables and AutoEdit members in Chapter 3.

Within CONTROL-M / Analyzer, AutoEdit members are available primarily for interaction with CONTROL-M functions.

One technical consideration for using AutoEdit members is their relationship to the PDS library and its naming conventions. Each AutoEdit member resides in a PDS library, and therefore has a member name. This name should coincide with your data center naming conventions. You may also want to follow the group naming conventions, discussed in Chapter 3.

Concurrent Usage and Locking Considerations

This section discusses the issues involved with concurrent usage and locking.

Concurrent Usage

Local variables, which are defined within each rule, are only usable within that one rule. By definition, local variables are never used concurrently by multiple rules or multiple users. This applies to all CONTROL-M / Analyzer variables (local, AutoEdit, and database).

Locking Considerations

CONTROL-M / Analyzer database variables are commonly used concurrently by multiple rules or users. For this reason, the CONTROL-M / Analyzer database has a locking capability. DO COMMIT parameters also affect the locking and unlocking of variables.

When considering locking, determine the answers to the following questions:

- Is it possible for any individual rule to lock a database variable for long periods of time?
- If so, will this cause a problem for other users and / or rules?
- Can the rule that needs to lock a variable for long periods of time be scheduled after hours, or at a time that does not coincide with the other users and / or rules?

A discussion of locking database variables can be found in the *CONTROL-M/Analyzer User Guide*. The user guide also includes a separate discussion of locking and unlocking database variables in relation to the DO COMMIT parameter. Among other topics regarding locking, the following is discussed in *CONTROL-M/Analyzer User Guide:*

- DO SET parameters that are used to specify database variables, should sometimes be located in a rule as close as possible to the corresponding DO COMMIT command. This may minimize the amount of time during which the database variable remains locked.
- DO COMMIT NOW parameters should be used to unlock database variables immediately, rather than waiting until the end of the rule.
- DO COMMIT DELETE parameters also unlock database variables so that they may be accessed by other rules. These parameters should be used as soon within in the rule when it will not be updating the database.

For example, in a balancing rule, there may be a long report being analyzed. As soon as he first intolerable error is found, the rule may be programmed not to update the CONTROL-M / Analyzer database. If any variables are locked at that time, then a DO COMMIT DELETE will free them for use by other users and / or rules.

- DO COMMIT NONE and DO COMMIT DELETE remove variables from the list of variables that will be updated. However, DO COMMIT NONE does not unlock the variables.

- Any termination of the rule will result in any locked database variables being unlocked.

Backing Up, Restoring, Restarting

The considerations for backup, restore, and restart of a CONTROL-M / Analyzer application, or of the entire CONTROL-M / Analyzer repository, are identical to those for any other database product.

Large installations usually have evolved backup and restore procedures for daily, weekly, monthly, and annual versions of their data and software. The procedures are sometimes specific to the software product and to the number of users involved, thus requiring a balance between the need for speedy recovery and the amount of time taken to perform a backup.

Within the CONTROL-M / Analyzer environment, data exists in a format understood by the INCONTROL products, for example, the Active Balancing file. There are also libraries, such as CONTROL-M / Analyzer JCL and RULES, which would likely be put into a standard system library. Because most installations already have backup procedures in place for standard system libraries, this discussion of backup focuses on the data specific to the CONTROL-M / Analyzer environment.

The repository discussed in the *CONTROL-M/Analyzer User Guide*, and in the CONTROL-M / Analyzer chapter of the *INCONTROL for z/OS Administrator Guide* describe the CONTROL-M / Analyzer and IOA files that are impacted by CONTROL-M / Analyzer rules and missions.

Included with CONTROL-M / Analyzer are a set of utilities which allow selective copying from CONTROL-M / Analyzer files. For more information on these utilities, refer to the CONTROL-M / Analyzer chapter in the *INCONTROL for z/OS Utilities Guide*. With the utilities provided, you have the ability to:

- copy specific variables or groups of variables
- copy invocation and report entries
- unload and load a CONTROL-M / Analyzer file to / from a sequential file

With these utilities, in combination with the IOA utilities provided for other IOA files, a backup and restore strategy can be designed which is suited to each balancing application.

Use of Utilities

CONTROL-M / Analyzer is delivered with a comprehensive set of utilities that assist within the production environment. When you are beginning to use CONTROL-M / Analyzer in production, there are a few utilities which are very important to be familiar with. These utilities are discussed below. For further information on these utilities, see the CONTROL-M / Analyzer chapter in the *INCONTROL for z/OS Utilities Guide*, and the CONTROL-M / Analyzer chapter in the *INCONTROL for z/OS Administrator Guide*.

Utility CTBJAFDL

This utility deletes unnecessary entries from the Job Activity and Report files. This utility should be added to the New Day procedure. For example, a parameter can be used to automatically delete all entries older than the specified number of days. The New Day procedure could be used automatically to delete all entries older than ten days. For further information see “Setting Up a New Day Procedure” on page 4-179.

By running this procedure daily, the Job Activity and Report files will be less likely to fill up to the limits of the space allocated.

Utility IOADBF

This utility is used to create and format various CONTROL-M / Analyzer files.

When installing CONTROL-M / Analyzer, this utility is activated to create the initial files. After installation, if the amount of user activity outgrows the original file allocations, this utility is used to create and format a new CONTROL-M / Analyzer file of a different size. It is most often used together with the CTB*xxx*CP utility to increase file size.

Utility CTBxxxCP

The value for *xxx* in the utility name can be one of the following:

- JAF – Job Activity and Report files
- DBV – Group and Database Variable files

This utility copies CONTROL-M / Analyzer data to other files. It is most often used in conjunction with utility CTBCRDF when expanding or shrinking the size of a file. Specific subsets of the files may be copied, or all of the entries may be copied.

Utility CTBFRM

Utility CTBFRM is used to compress the Active Balancing file on a daily basis. For a discussion of this topic and cleanup utilities, See “Setting Up a New Day Procedure” on page 4-179.. For further information on this utility, see the CONTROL-M / Analyzer chapter in the *INCONTROL for z/OS Utilities Guide*, and the CONTROL-M / Analyzer chapter in the *INCONTROL for z/OS Administrator Guide*.

Setting Up a New Day Procedure

Purpose

The New Day Procedure is the primary mechanism used to order, or activate, new missions for CONTROL-M / Analyzer. It is also used to perform daily cleanup of the CONTROL-M / Analyzer Jobs Activity and Reports files, to condense the Active Balancing file, and to backup or unload files.

For further information on the New Day procedure, see the CONTROL-M / Analyzer chapter in the *INCONTROL for z/OS Utilities Guide*, and the CONTROL-M / Analyzer chapter in the *INCONTROL for z/OS Administrator Guide*.

In conjunction with your first balancing application going into production, the New Day Procedure is implemented. This is usually done by the systems administrator responsible for CONTROL-M / Analyzer implementation. If the first balancing application does not use missions, a New Day Procedure would still be implemented to perform daily cleanup, as noted above.

Cleanup Activities

Every time a CONTROL-M / Analyzer mission or rule is activated,

- an entry is made in the Job Activity file and the Report file
- with the activation of a mission, an entry is made in the Active Balancing file

While individual entries may take up little disk space, many entries can be made during a day because many rules and missions are invoked. Additionally, users may concurrently be testing new applications, rerunning rules many times during a day. Further, even one user report can be very large if a rule was executed with a very large input file.

If entries are never deleted from these files, eventually the files will fill up and no new entries would be permitted to the files.

To avoid such unexpected surprises, the New Day Procedure initiates automatic cleanup procedures. These procedures are part of the default New Day Procedure to

- back up the Active Balancing file
- delete unnecessary entries from the Active Balancing file, and compress the file, as discussed in “Utility CTBFRM” on page 4-179

Additionally, you may want to

- add a step to the New Day Procedure that instructs utility CTBJAFDL to delete unnecessary entries from the Jobs Activity and Report files, as discussed in “Utility CTBJAFDL” on page 4-178
- include the running of other supplied utilities to check the integrity of the various CONTROL-M / Analyzer files

Missions

Any rule that requires the use of scheduling parameters may require a mission. The user designing the rule and the mission would inform the system administrator of the need to update the New Day Procedure to add an order request for that mission.

Glossary

Table 7 Glossary of Terms (part 1 of 4)

Term	Definition
Active Balancing File	A file containing entries for all missions that are likely to execute in the very near future. It is a control file used by operations personnel to query and update the status of the different balancing tasks.
Audit Trail	An IOA Log file containing a complete audit trail of every event under the CONTROL-M / Analyzer balancing environment. CONTROL-M / Analyzer logs every item of meaningful information about its operation.
AutoEdit Variables	A set of one or more variables, stored in a PDS library member. These variables can be dynamically read and updated during rule execution. They may be accessed by other INCONTROL products.
Balancing Mission	A set of balancing rule scheduling criteria and other parameters, stored in a PDS library member.
Balancing Status	The current situation of active balancing tasks, such as awaiting execution, balancing, ENDED OK, ENDED NOTOK).
Block	See ON Block.
Category	One set of scheduling criteria, forming part of a mission. A single mission may contain any number of categories.
Commit	The action taken to update the CONTROL-M / Analyzer database with new values of one or more variables.
Conditions	Prerequisite conditions, stored in the IOA Conditions file. These conditions can be added or removed by a CONTROL-M / Analyzer rule based on the results of rule execution. A prerequisite condition, which must exist prior to rule execution, can also be specified by a mission.
CONTROL-D	The Output Management System. Automatically schedules and controls every aspect of report processing and distribution, including report decollating, bundling, printing, online viewing, and archiving.
CONTROL-M	The Automated Production Control and Scheduling System. Manages and automates the setup, scheduling and execution of jobs in the OS / 390 data center.

Table 7 Glossary of Terms (part 2 of 4)

Term	Definition
CONTROL-O	The Console Automation System and Desired State Monitoring System. Monitors and automatically responds to messages, commands, and dataset events, as well as various other system events. The CONTROL-O / COSMOS feature allows for status monitoring while maintaining all critical system objects in a desired and ideal status.
CONTROL-M / Restart	The Restart Management System. Automates the activities that must be performed when restarting failed jobs, including the scratching and uncataloging of datasets created by failed jobs.
Database (CONTROL-M / Analyzer)	A set of files that keeps track of information about CONTROL-M / Analyzer Database variables, their generations of values and the groups to which they belong.
DO action	A balancing parameter used in a rule, which specifies a balancing action to take place (for example, print a line in the balancing report, set a variable to a value, terminate a rule).
Editor (CONTROL-M / Analyzer)	A facility that provides line editing operations (moving, copying, deleting, repeating) within the Rule Definition facility of CONTROL-M / Analyzer.
CONTROL-M Application Server (CTMAS)	A workstation implementation of the IOA environment, providing one focal point for enterprise production activity.
External Data Source	A file or report that a CONTROL-M / Analyzer rule can access to perform its balancing function. The file may be a sequential file or a compressed CONTROL-D output file.
Extract data	A function performed within a rule, using DO EXTRACT parameters that bring data into the CONTROL-M / Analyzer environment from external files or reports.
Functions	A set of facilities that can be used within a rule to process numeric and string values, such as absolute value, square root, substring, integer.
Generation	A value of a variable, associated with a specific update cycle, such as date and time.
Group	A parameter specified in CONTROL-M / Analyzer rules, missions, and Database Variable definitions, and used in various online displays. It is used to provide a logical association between data with a common purpose, within the IOA repository.
INPUT block	One of the two types of ON blocks used within a rule to specify the characteristics and extraction criteria for external data sources.
Invocation Report	A CONTROL-M / Analyzer report, produced as a result of a rule execution, tracing all of the steps that occurred from the time the rule was invoked, until it terminated.
IOA	Integrated Operations Architecture is the software architecture of the INCONTROL family of automated systems operations products.
JCL	Job Control Language. This is the language used to communicate with the operating system of the computer.

Table 7 Glossary of Terms (part 3 of 4)

Term	Definition
Job Activity	A file containing historical information about balancing activity and roll-backs within jobs. For example, the file could contain names of variables committed, reference to reports generated, balancing results.
Label	A name given to one section of an ON block, within a rule.
Level	A reference to specific lines in a report or file, directly associated with a WHEN statement within a rule.
Local Variables	Variables used within a rule, which are not saved. These variables are known only to the rule as it executes.
Locking	An action automatically taken on CONTROL-M/Analyzer database variables to prevent more than one user from accessing or updating a variable at the same time.
Log	An IOA Log file containing a complete audit trail of every event under the CONTROL-M/Analyzer balancing environment. CONTROL-M/Analyzer logs every item of meaningful information about its operation.
Management by Color	An approach to the use of colors on INCONTROL product displays, used by INCONTROL products to assist operations, systems, and user staff in finding critical information on screens.
Mission	See Balancing Mission.
MODE	A parameter specified in an ON statement within a rule. It indicates whether a file or report is to be treated as a page or as a data set.
Multi-Screen Control	An IOA facility that enables a user to jump from one function to another, at will, and to return to the exact point at which the user was located in a previous function.
New Day Procedure	A set of jobs, set up by the system administrator, to order, or activate, new missions, and to perform daily clean-up functions.
ON Block	A logical section of a rule, containing an ON statement and various balancing parameters. ON blocks may operate on external data (files or reports) or internal data.
ON Data	One of the two types of ON blocks, used within a rule, to specify balancing actions associated with variables already within the CONTROL-M/Analyzer environment.
Owner	The user ID associated with the creation of a rule, mission or variable.
Rollback facility	A facility provided within CONTROL-M/Analyzer to automatically reset CONTROL-M/Analyzer Database variables to their values at an earlier point in time.
Row	One line of a file or report.
Rule	A set of automated balancing parameters, performing one logical balancing function and stored under one name in a PDS library.
Search criteria	A set of parameters within a WHEN statement of a rule, that specifies how to find data in a file or report.
SHOUT	An action parameter within a rule that sends a message to an online user.
SHOW	A command, used within the CONTROL-M/Analyzer online environment, to show options used for displaying information.

Table 7 Glossary of Terms (part 4 of 4)

Term	Definition
String	A set of one or more characters.
SYSOUT	The symbolic name that CONTROL-M/Analyzer and the operating system use to refer to a printed report.
SYSPRINT	The symbolic name that CONTROL-M/Analyzer and the operating system use to refer to the Invocation Report.
SYSRULE	The symbolic name that CONTROL-M/Analyzer and the operating system use to refer to a listing of the rule.
SYSUSER	The symbolic name that CONTROL-M/Analyzer and the operating system use to refer to the balancing report (meaning, all of the print lines resulting from DO PRINT parameters specified within a rule).
TERMINATE	A balancing action, specified within a rule, that causes a rule to end.
User Code	A numeric code, specified by a user when a rule terminates, to add significance to the results of a rule execution, for operators and/or users.
WHEN	A statement found within a rule that specifies search criteria used to find data in reports or files.
WHEN statement	A statement used in a CONTROL-M/Analyzer rule, which specifies the search criteria to find data in a report or file.

Index

A

Active Balancing Environment 52
Advanced Problems and Solutions 108
Analyze Unexpected Results 162
AutoEdit Members 175

B

Backing Up, Restoring, Restarting 177
Balancing Mission 45
Basic Problems and Solutions 74
Before You Begin 16
BMC Software, contacting 2
Build and Test your Rule Piece by Piece 161

C

Cleanup Activities 180
Concurrent Usage 176
Concurrent Usage / Locking Considerations 176
CONTROL-D / WebAccess Server
    Product description 152
CONTROL-M / Analyzer and IOA Fundamentals 174
CONTROL-M / Analyzer
    Product Description 152
CONTROL-O
    Product Description 153
CONTROL-O / COSMOS
    Product Description 153
CONTROL-D
    Product Description 152
CONTROL-D / Image
    Product Description 153
CONTROL-D / Page On Demand
    Product Description 152
CONTROL-M
    Product Description 152
CONTROL-M / Tape
    Product Description 152
CONTROL-M / Restart
    Product Description 152
CONTROL-V
    Product Description 152
Conventions Used in This Guide 16
customer support 3

D

Daily Invoice File 72
Data in Different Locations in a Report 158
Data that Must Be Associated with External Data before Balancing 158
Database Variable Worksheet 171
Database Variables 24

F

Finding Data in Reports and Files 157
For Newcomers to INCONTROL Products and the IOA Environment 150

G

GROUPS, Defined by Database Organization 154
GROUPS, Defined by Functional Responsibility in the Work Place 154
GROUPS, Defined by MIS Application Area 154

I

Implementation Hints 161
INCONTROL Products 152
Integrated Activity Log 151
Integrated Management Reporting 151
INTRAC's Problems — General 71
Invoking the CONTROL-M / Analyzer Environment 166
IOA Conditions / Resources Screen 58
IOA Log 63

K

Keep Separate Libraries for Production and Testing 157

L

Locking Considerations 176

M

Mission Worksheet 170
Missions 180

N

Naming Groups 153
Naming Variables 154

O

Organize Libraries by Group / Application 156
Organizing Libraries 156

P

Preparation 23
Problem 103, 108, 125, 132
Problem 1
    The Daily Invoice File — Warning 74
Problem 2
    Check Payroll Totals 90
Problem 3
    Check Sales Commissions 103
Problem 4
    Balance Divisional Accounting Files 108
Problem 5
    Check Weekly Database Amounts 124
Problem 6
    Balance Monthly Commissions 132
Problems 90
Product Description
    CONTROL-M / Restart 152
    CONTROL-M / Tape 152
    CONTROL-O 153
    CONTROL-O / COSMOS 153
    CONTROL-D 152
    CONTROL-D / Image 153
    CONTROL-D / Page On Demand 152
    CONTROL-M 152
    CONTROLM / Analyzer 152
    CONTROL-V 152
product support 3
Purpose 179

Q

Quality Control and Avoiding Errors 164

R

Research before Online Definition 161
Rule Activity 38
Rule Definition 161
Rule Definition Summary 145
Rule Template
    Worksheet A 168
Rule Worksheet B 169
Rules 33

S

Sales Commission Detail Report 73
Saving Data in AutoEdit Members 160
Saving Data in the CONTROL-M / Analyzer Database 160
Setting Up a New Day Procedure 179
Short ON Blocks vs. Long ON blocks 163
Solution 74, 92, 104, 108, 126, 132
Solution — Overview 91, 103, 108, 125, 132
Solution 1 — Overview 74
Sources of Data — Reports and Files 72
Structured Approach for Implementation 167
Suggested Conventions 153
support, customer 3
SYSPRINT — Invocation Report 88
SYSRULE Report 89

T

technical support 3
The Example Company, INTRAC 68
The rule consists of four ON blocks
    132

U

Use of Utilities 178
Using CONTROL-M / Analyzer's Rule Definition Editor
    vs. the System Editor 164
Using the NOT EQUAL Condition to Identify Data 157
Utility CTBFRM 179
Utility CTBJAFDL 178
Utility CTBxxxCP 179
Utility IOADBF 178

V

VARIABLES, Associated with an “Owner” 156
VARIABLES, With a Prefix or Suffix of a
    Division / Department Code 155

Notes

62363