CONTROL-O Solveware Reference Guide

Supporting

CONTROL-O Solveware version 6.2.21

July 2007

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

Restricted rights legend

Customer support

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

Support website

You can obtain technical support from BMC 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 offers
- find the most current information about BMC products
- search a database for issues similar to yours and possible solutions
- order or download product documentation
- report an issue or ask a question
- subscribe to receive proactive e-mail alerts when new product notices are released
- find worldwide BMC 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 customer_support@bmc.com. (In the subject line, enter **SupID**:<***yourSupportContractID***>, such as SupID:12345). Outside the United States and Canada, contact your local support center for assistance.

Before contacting BMC

Have the following information available so that Customer Support can begin working on your issue 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 issue
- 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

Tables

About This Guide

This guide contains console automation solutions that enable easy and rapid implementation of CONTROL-O for a high degree of console automation.

Chapter 1–Introduction

Overview of CONTROL-O SolveWare, introducing key concepts in solution logic and providing a general description of the format of SolveWare documentation.

Chapter 2–Solveware Subject STARTSYS

Solutions for automating system startup and initialization.

Chapter 3–Solveware Subject SHUTSYS

Solutions to automate system shutdown.

Chapter 4–Solveware Subject SUPPRESS

Solutions for suppression of certain messages (for MVS, JES2, JES3, and system components).

Chapter 5–Solveware Subject DEVICES

Solutions for handling of device status and CONTROL-M resources.

Chapter 6–Solveware Subject LOGREC

Solutions for automation of interaction with the Environmental Record Editing and Printing (EREP) facility.

Chapter 7–Solveware Subject SMF

Solutions for automation of interaction with SMF.

Chapter 8–Solveware Subject DUMPDS

Solutions for automation of management of system dump datasets.

Chapter 9–Solveware Subject CONSOLES

Solutions for automation of management of WTO (Write to Operator) message buffers.

Chapter 10–Solveware Subject TSO

Solutions for automation of interaction with a TSO environment.

Chapter 11–Solveware Subject CICS

Solutions for automation of interaction with CICS.

Chapter 12–Solveware Subject DB2

Solutions for linking DB2 status with CONTROL-M jobs.

Chapter 13–Solveware Subject IMS

Solutions for automation of interaction with IMS.

Chapter 14–Solveware Subject OMEGAMON

Solutions to utilize problem detection and resolution capabilities of OMEGAMON.

Chapter 15–Solveware Subject TESTJOBS

Solutions for handling of certain error situations that can result from non-production jobs.

Chapter 16–Solveware Subject ADABAS

Solutions for automation of interaction of CONTROL-M with ADABAS.

Chapter 17–Solveware Subject IDMS

Solutions for automation of interaction of CONTROL-M with IDMS.

Chapter 18–Solveware Subject NETVIEW OPEN ACCESS

Solutions for automation of interaction of CONTROL-O with NetView.

Chapter 19–Solveware Subject CTORMT

Solutions for automation of operations on remote systems.

Chapter 20–Solveware Subject TCP/IP

Solutions for facilitation of file transfers between various machines connected by a TCP / IP network.

Chapter 21–Solveware Subject INITIATORS

Solutions for maximizing initiator utilization by CONTROL-M.

Chapter 22–Cross-Reference Lists

Contains tables listing the event, command, script, and message cross-reference lists.

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.

Information New to This Version

Where substantive additions and modifications to the content of this guide occur, revision bars have been inserted in the margin.

Related Publications

CONTROL-O Conversion Guide

Concepts and steps required for converting from other console automation products to CONTROL-O.

CONTROL-O User Guide

A complete guide to CONTROL-O features, options, and usage.

CONTROL-O/COSMOS User Guide

A complete guide to CONTROL-O / COSMOS features, options and implementation.

CONTROL-O/Server User Guide

A complete guide to CONTROL-O / Server features, options and usage.

INCONTROL for z/OS Administrator Guide

Information for system administrators about customizing and maintaining INCONTROL™ family of products.

INCONTROL for z/OS Installation Guide

A step-by-step guide to installing INCONTROL family of products using the INCONTROL™ Installation and Customization Engine (ICE) application.

INCONTROL for z/OS Messages Manual

A comprehensive listing and explanation of all INCONTROL messages and codes.

INCONTROL for z/OS Security Guide

A step-by-step guide to implementing security in INCONTROL family of products using the INCONTROL Installation and Customization Engine (ICE) application.

INCONTROL for z/OS Utilities Guide

Describes utilities designed to perform specific administrative tasks that are available to INCONTROL products.

Chapter 1

Introduction

SolveWare Solutions are composed of the following major components:

- SolveWare solution documentation – This guide provides documentation for the SolveWare Solution Kit. It includes detailed descriptions of the automation solutions that are easy to understand and easy to implement.
- SolveWare message management solutions – The CONTROL-O installation tape includes a library of predefined automation solutions that handle many of the most frequently occurring messages of MVS, VTAM, JES, CICS, and so on. At most sites, these solutions can be applied with little or no modification.
- SolveWare advanced automation solutions – The CONTROL-O installation tape includes a library of predefined automation solutions that utilize KeyStroke OpenAccess (KOA) and NetView OpenAccess (NOA) scripts.
- Cross reference lists – A separate list is provided for each of the following: message, command, event and script names. Each message, command, event, or script name referenced by SolveWare is listed. This is followed by the SolveWare table in which the name and the relevant documentation page numbers are listed.

NOTE

CONTROL-O / COSMOS is an integral part of CONTROL-O that performs the same functions as many rules described in this guide. This is especially true for SolveWare rules designed to handle startup and termination of various objects or resources in your computing environment.

If CONTROL-O / COSMOS has been installed and implemented at your site, ensure that redundant rules are no longer loaded.

SolveWare Implementation Considerations

This section describes the steps involved in implementing SolveWare solutions.

Step 1. Studying Solveware Documentation and Definitions

The first step in implementing solutions is to examine the solution definitions that are provided in this guide to determine if each subject is relevant, or if the automation implementation concept is applicable, to your site.

Documentation Format

The SolveWare documentation is divided into the following parts:

- SolveWare subjects

  Groups of solutions organized by common problem category, such as CICS, DB2, or SUPPRESS. Each SolveWare subject is described in a separate chapter in this guide.

- Solutions

  Resolutions to specific problems. Solutions are made up of one or more rules and may also utilize KOA scripts, CONTROL-M jobs, and so on.

- Rules

  CONTROL-O rule definitions used to solve specific automation problems.

- KOA scripts

  KeyStroke OpenAccess scripts are invoked by rules. Transcripts interface with VTAM applications such as OMEGAMON to solve specific automation problems.

Each SolveWare subject provides one or more solutions and each solution consists of one or more rules.

Terminology

Rule documentation in this guide refers to the following rule classifications in CONTROL-O:

- Scheduled rule

  Rule that was loaded into memory, automatically or manually.

- Active rule

  Rule for which all runtime scheduling criteria have been satisfied.

  In the CONTROL-O Rule Status screen, an active rule has an ACTIVE status.

- Inactive rule

  Rule for which runtime scheduling criteria have not yet been satisfied.

  In the CONTROL-O Rule Status screen, an inactive rule has a WAIT ACTIVATION status.

- Triggered rule

  A rule is triggered when its ON statement criteria have been satisfied (for example, the relevant command or message is issued). Event rules are triggered when they become active and can be triggered again cyclically.

Solution Definitions

The library and specific tables in which the rule is found are indicated in the documentation of the rule. Depending on whether the rule triggers jobs or KOA scripts, parts of solutions can be found in any of the appropriate CONTROL-O SolveWare libraries:

Table 1 SolveWare Libraries

Library	Description
SOLVRULE	Sample CONTROL-O rule tables.
SOLVJCL	Sample JCL members of CONTROL-M jobs that are triggered by rules.
SOLVSCHD	Sample scheduling definitions for CONTROL-M jobs that are triggered by rules.
SOLVKOA	Sample KOA scripts that are triggered by rules.

STARTSYS Rules

SolveWare rules provided with CONTROL-O assume that certain Global variables (with specific values) and certain prerequisite conditions exist at your site.

To ensure that these variables and conditions exist and are compatible with SolveWare rule definitions, the rules included in SolveWare subject STARTSYS must be implemented.

Step 2. Customizing Rules Before Implementing

Rules may need to be tailored to the environment. Rule documentation contains customization considerations and instructions.

SolveWare Categories

The SolveWare category specified in the rule documentation provides an estimate of the amount of customization required. Rules are categorized according to the type of customization that the user must perform in order to adapt the rule to the environment.

SolveWare categories are:

Table 2 SolveWare Categories

Category	Description
Category 1	Minimal or no customization required.
Category 2	Some customization required.
Category 3	The rule is provided as an example that may or may not be applicable to the site. The amount of customization required is dependent upon site requirements.

Dynamic Destinations

SolveWare solutions send SHOUT messages to various destinations that are not necessarily fixed. Such destinations are known as dynamic destinations and are maintained in the IOA Dynamic Destination table. This table enables the user to specify a group name destination and the final multiple destinations it represents.

SHOUT destinations that are defined in the IOA Dynamic Destination table can be updated in the table instead of updating the SHOUT destinations individually in the rule definitions.

SolveWare SHOUT messages are sent to the following dynamic destinations:

Table 3 Destinations for SHOUT Messages

Message	Description
U-IOADMIN	INCONTROL administrator
U-SYSADMIN	System administrator
U-SYSCICS	CICS administrator
U-SHFTOPER	Shift operator
U-SYSDBA	Database administrator

OWNER Parameter

Parameter OWNER is often used to determine whether a rule has authorization for requested actions. Customize this parameter according to your on-site security standards so that SolveWare rules are granted required authorization.

Inverse IN Conditions

Many SolveWare solutions use the CONTROL-O inverse IN condition feature to deactivate rules. The use of inverse IN conditions greatly simplifies rule deactivation.

Inverse IN conditions are indicated by a “Not” symbol (¬ or !) prefixed to the condition. For example: `!CTO-IFB060E-HANDLED`

A rule defined with an inverse IN condition is inactive when its IN condition does exist and active when its IN condition does not exist. For more details, see the rule parameters chapter of the *CONTROL-O User Guide*.

It is also possible to use a standard (non-inverse) prerequisite IN condition to deactivate rules. Such a rule is inactive when its IN condition does not exist and active when its IN condition does exist. However, before implementing such a rule, the required IN condition must be added manually in order to activate the rule for the first time.

For this reason, SolveWare solutions use inverse IN conditions that do not require adding the condition manually before implementing the rule.

Rule Thresholds

The term “threshold” refers to the maximum number of times a message can appear within a specified number of minutes. The threshold parameters for each message can be adapted to site requirements.

Following are some instances where you should specify a threshold.

Example 1

A rule “shouts” to a TSO user whenever a specific message appears on the console. For some reason, this message is issued many times and the TSO user is notified repeatedly. The messages are the result of a single problem and the information they provide repeatedly is therefore irrelevant.

In this case, it is recommended to stop the automatic response to the repetitive console messages after a threshold is reached.

Example 2

A rule triggers a job that copies and cleans SYS1.LOGREC whenever a specific message appears on the console. For some reason, SYS1.LOGREC fills up very quickly and the SYS1.LOGREC dataset is dumped repeatedly.

In this case, it is recommended to stop dumping the dataset and send a message to the system administrator after a threshold is reached. The person responsible can then handle any hardware problem or, if necessary, resize the SYS1.LOGREC dataset.

Step 3. Testing and Implementing Rules

Test all SolveWare rules before they are implemented. The recommended method for testing rules is to trigger a rule operating in TEST or LOG mode using utility IOATEST (detailed below). Next, review the test journal (log) of the rule’s events and actions. After all events and actions are satisfactory, the rule can be scheduled for execution in the production mode.

Operation Modes

The Recommended Mode in the rule documentation refers to the mode for initial testing and/or operating. The following table describes the operation modes.

Table 4 Operation Modes

Mode	Description
TEST	Console actions are not performed; they are written to a test journal. Setting of Global variables is performed.
LOG	The rule is processed normally (as in PROD mode) and, in addition, all identified events and actions are written to a test journal.
PROD	Production mode. The rule is processed normally.

NOTE

Before operating rules in LOG or PROD mode, remove any previously used parallel automation mechanism.

For example, if SMF Exit IEFU29 has been implemented in order to dump switched SMF datasets, remove the exit prior to operating the SolveWare rule that dumps switched SMF datasets.

Problem Simulation

In order to test rules, rules must be triggered utility IOATEST can be used to issue messages to the console subsystem, thus simulating the problem situation and triggering the relevant rules.

For example, message `IEF281I ddd NOW OFFLINE` can be sent to the console subsystem using utility IOATEST without actually varying the device off-line. The relevant rule is triggered by this message and its results can then be examined.

For license and warranty information, refer to member $$RULES in the -O Installation library.

STARTSYS

SolveWare subject STARTSYS automates the initialization of SolveWare rules that handle the startup of various system components after IPL.

Solutions Provided

SolveWare subject STARTSYS contains the following solutions:

- SolveWare initialization – This solution contains rules that initialize prerequisite conditions and Global variables required by other SolveWare rules at time of IPL.
- System startup – This solution contains sample rules for automated system startup.

SolveWare Initialization

This solution automates SolveWare rule initialization by adding or deleting those prerequisite conditions that must be added or deleted at time of IPL and by setting certain Global variables to values required by SolveWare rules.

The SolveWare Initialization solution is implemented as a prerequisite to the implementation of all other SolveWare solutions. For special administrative considerations regarding SolveWare initialization, see the customization instructions for rule CTO147I, in Table 5 on page 33.

Rules

The SolveWare Initialization solution includes the following rules:

- IPL – Reset SolveWare Prerequisite Conditions
- Define Site Configuration Global Variables

Rules Structure

The following tables describe the structures of the SolveWare Initialization solution rules.

Table 5 IPL – Reset Prerequisite Conditions Rule Structure (part 1 of 2)

Item	Description
Title	IPL – Reset SolveWare Prerequisite Conditions
Name	CTO147I
Table	INITSLV
Message	`CTO147I CONTROL-O INITIALIZATION COMPLETE. TYPE=`*`type`*`,SUB=`*`subsystem`*`,RUN#=`*`nnnn`*
Message Description	This rule is triggered immediately following the startup of the CONTROL-O monitor during IPL. It performs actions needed to initialize other SolveWare rules. This initialization process consists of adding or deleting prerequisite conditions for other rules that must be added or deleted at time of IPL.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	This rule is automatically triggered immediately following startup of CONTROL-O during IPL, before any other SolveWare rules are scheduled. This rule adds or deletes prerequisite conditions required by other rules. To determine which prerequisite conditions this rule must add or delete, refer to the description of each SolveWare rule in this guide.
Rule Actions	Adds or deletes prerequisite conditions required by other rules.
Activating the Rule	The rule is triggered immediately following the startup of the CONTROL-O monitor during IPL.

Table 5 IPL – Reset Prerequisite Conditions Rule Structure (part 2 of 2)

Item	Description
Recommended Mode or Category	During the testing period, the rule is activated in LOG mode. Once you are satisfied with the results of the rule, change this mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.

Customization	The rule adds or deletes conditions for all SolveWare rules included in this guide. Decide which SolveWare subjects (and rules) you intend to implement and then remove the conditions for those rules you do not intend to implement.
	
	Start the CONTROL-O monitor at the beginning of IPL. This is achieved by placing the following command as one of the first commands in member COMMND*nn* in SYS1.PARMLIB:
	
	`COM='S CONTROLO,SUB=MSTR,OUTPUT=DUMMY,`
	
	`ORDER=IPLRULES,TYPE=IPL'`
	
	Rule list member IPLRULES contains table INITSLV. Optionally, it can contain system startup rules (table STARTSYS) and started task monitoring rules.
	
	For example, tables CICS and DB2.

Table 6 Define Site Configuration Global Variables Rule Structure (part 1 of 3)

Item	Description
Title	Define Site Configuration Global Variables
Name	JES2GLBL
Table	INITSLV2
Event	JES2GLBL
Event Description	This Event rule is triggered immediately following the startup of the CONTROL-O monitor during IPL. It initializes site configuration Global variables that define the MVS multi-system configuration to CONTROL-O. The Global variables defined by this rule are necessary for the implementation of other SolveWare rules at your site.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Table 6 Define Site Configuration Global Variables Rule Structure (part 2 of 3)

Item	Description

Global Variables	■ %%ID_*sfmid*
	
	Character used by CONTROL-M as a system identifier (set in AutoEdit System variable %%$SIGN) in a multi-system environment, where *smfid* is the SMF ID of the CPU.
	
	For example:
	`%%ID_SYS1=A`
	`%%ID_ESA1=B`
	
	■ %%JES2_ROUTE_*smf*
	
	Multi-access spool member, multi-access spool member node, or node member used to send a command over the network to the specified system. *smf* is the SMF ID of the CPU.
	
	For example:
	
	`%%JES2_ROUTE_ESA1=$M2 %%JES2_ROUTE_ESA2=$N2M3`
	
	■ %%CPU_SMF*x*
	
	SMF ID of the CPU, indexed by a sequence number.
	
	For example:
	
	`%%CPU_SMF0=SYS1`
	`%%CPU_SMF1=SYS2`

Rule Logic	This Event rule is executed at start of CONTROL-O to initialize JES2 Global variables that define the MVS multi-system configuration to CONTROL-O. These variables are used by other SolveWare rules.
Rule Actions	Initialize Global variables %%ID_*smfid*, %%JES2_ROUTE_*smf* and %%CPU_SMFx for each MVS system handled by CONTROL-O.

Table 6 Define Site Configuration Global Variables Rule Structure (part 3 of 3)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	For each MVS system handled by CONTROL-O, a set of Global variables must be defined in the rule as follows: ■ %%ID_*smfid* Character used by CONTROL-M as a system identifier (set in AutoEdit System variable %%$SIGN) in a multi-system environment). ■ %%JES2_ROUTE_*smf* Multi-access spool member, multi-access spool member node, or node member used to send a command over the network to the specified system. *smf* is the SMF ID of the CPU. For example: `%%JES2_ROUTE_ESA1=$M2 %%JES2_ROUTE_ESA2=$N2M3` ■ %%CPU_SMF*x* SMF ID of the CPU, indexed by a sequence number.

System Startup

This solution contains sample rules for automated system startup using CONTROL-O. The startup process is initiated automatically after IPL. A statement in member COMMNDnn in the SYS1.PARMLIB library starts the IPL CONTROL-O. STARTSYS rules start the various system components in the required sequence (for example, start TSO and CICS only after VTAM node initialization is completed) and reply to the appropriate JES2 startup messages.

Solution “System Startup” does not contain a rule to start JES2 because it assumes that JES2 is started automatically after IPL. If desired, you can define a rule to start JES2, but first, prevent automatic startup of JES2 outside of CONTROL-O.

NOTE

Initialize (delete) all prerequisite conditions defined in this solution before the rules of this solution are ordered. For special considerations regarding prerequisite condition initialization, see “SolveWare Initialization” on page 31.

Deactivate any existing startup mechanisms that parallel the rules in this solution.

This solution handles startup of JES2 environments. However, the solution can be easily adapted to handle startup in a JES3 environment.

Rules

The System Startup solution includes the following rules:

- JES2–Specify Options
- Start VTAM
- TSO VTAM Node Is Active
- CICS VTAM Node Is Active
- Start New CONTROL-O Under JES2
- Start CONTROL-M
- Start CONTROL-D Monitor and Subsystem
- Order Non-IPL Rules

Rules Structure

The following tables describe the structures of the System Startup solution rules.

Table 7 JES2–Specify Options Rule Structure (part 1 of 2)

Item	Description
Title	JES2–Specify Options
Name	$HASP426
Table	STARTSYS
Message	`$HASP426 SPECIFY OPTIONS - subsystem`
Message Description	JES2 requests that the operator specify initialization options.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	The rule notifies the operator that after a predefined delay, CONTROL-O automatically replies NOREQ to the message. The operator can override the anticipated NOREQ option by replying to the message during the delay period.
Rule Actions	Notifies the operator that reply NOREQ is issued after a delay. After the delay, issues the following command: R %%REPLY,NOREQ

Table 7 JES2–Specify Options Rule Structure (part 2 of 2)

Item	Description
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	Activate the rule in LOG mode during the testing period. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of this rule requires some customization.
Customization	STARTSYS rules are provided as guidelines for automated system startup. Examine and adapt each rule to site requirements. The rule is defined with a 20-second delay period. This delay period (defined in parameter WAIT of statement DO COMMAND) can be adapted to site requirements. The reply string, originally defined as NOREQ, can be adapted to site standards.

Table 8 Start VTAM Rule Structure (part 1 of 2)

Item	Description
Title	Start VTAM
Name	SVTAM
Table	STARTSYS
Event	SVTAM
Event Description	This Event rule starts VTAM.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	This rule issues a START command to start VTAM. VTAM remains in status STARTING until JES is up, at which point VTAM starts executing.
Rule Actions	Issues the following command `S VTAM`
Activating the Rule	The rule is activated immediately, once it has been ordered.

Table 8 Start VTAM Rule Structure (part 2 of 2)

Item	Description
Recommended Mode or Category	Activate the rule in LOG mode during the testing period. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of this rule requires some customization.
Customization	STARTSYS rules are provided as guidelines for automated system startup. Examine and adapt each rule to site requirements. Change the VTAM procedure name specified in the rule (VTAM) to the actual name of the VTAM procedure at the site.

Table 9 TSO VTAM Node Is Active Rule Structure (part 1 of 2)

Item	Description
Title	TSO VTAM Node Is Active
Name	IST093I
Table	STARTSYS
Message	`IST093I` *`nodename`* `ACTIVE` where *nodename* is TSOM.
Message Description	The VTAM node specified in the message was successfully activated.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	Once the TSO VTAM node is successfully activated, TSO can be started. The rule immediately issues command S TSO to start TSO. Prerequisite condition or date CTO-STRT-TSO-ISSUED 0101 is added. This condition is defined as an IN condition for rule ORDERALL (see Table 14 on page 42).
Rule Actions	Issues the following command: `S TSO` Adds prerequisite condition or date CTO-STRT-TSO-ISSUED STAT
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.

Table 9 TSO VTAM Node Is Active Rule Structure (part 2 of 2)

Item	Description
Recommended Mode or Category	Activate the rule in LOG mode during the testing period. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of this rule requires some customization.
Customization	STARTSYS rules are provided as guidelines for automated system startup. Examine and adapt each rule to site requirements. Adapt the TSO VTAM node name specified in the rule (TSOM) to actual site VTAM definitions.

Table 10 CICS VTAM Node Is Active Rule Structure (part 1 of 2)

Item	Description
Title	CICS VTAM Node Is Active
Name	IST093I
Table	STARTSYS
Message	`IST093I` *`nodename`* `ACTIVE` where *nodename* is `CICS`.
Message Description	The VTAM node specified in the message was successfully activated.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	Once the CICS VTAM node is successfully activated, CICS can be started. The rule immediately issues command S CICS to start CICS. Prerequisite condition or date CTO-STRT-CICS-ISSUED 0101 is added. This condition is defined as an IN condition for rule ORDERALL (see Table 14 on page 42).
Rule Actions	Issues the following command `S CICS`. Adds prerequisite condition or date `CTO-STRT-CICS-ISSUED STAT.`
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.

Table 10 CICS VTAM Node Is Active Rule Structure (part 2 of 2)

Item	Description
Recommended Mode or Category	Activate the rule in LOG mode during the testing period. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of this rule requires some customization.
Customization	STARTSYS rules are provided as guidelines for automated system startup. Examine and adapt each rule to site requirements. Adapt the CICS VTAM node name specified in the rule (CICS) to actual site VTAM definitions.

Table 11 Start New CONTROL-O Under JES2 Rule Structure

Item	Description
Title	Start New CONTROL-O Under JES2
Name	SCTO
Table	STARTSYS
Event	SCTO
Event Description	This Event rule starts CONTROL-O under JES2.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	The rule issues a START command to start a new CONTROL-O monitor. The new CONTROL-O monitor remains in status STARTING until JES is up, at which point it starts executing and replaces the active CONTROL-O monitor. CONTROL-O is started under JES to enable it to write to SYSOUT files.
Rule Actions	Issues the following command `S %%CONTROLO.`
Activating the Rule	The rule is activated immediately, once it has been ordered.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of this rule requires some customization.
Customization	STARTSYS rules are provided as guidelines for automated system startup. Examine and adapt each rule to site requirements.

Table 12 Start CONTROL-M Rule Structure

Item	Description
Title	Start CONTROL-M
Name	SCTM
Table	STARTSYS
Event	SCTM
Event Description	This Event rule starts CONTROL-M under JES2.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	The rule issues a START command to start CONTROL-M. CONTROL-M remains in status STARTING until JES is up, at which point CONTROL-M starts executing.
Rule Actions	Issues the following command: `S CONTROLM`
Activating the Rule	The rule is activated immediately, once it has been ordered.
Recommended Mode or Category	Activate the rule in LOG mode during the testing period. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of this rule requires some customization.
Customization	STARTSYS rules are provided as guidelines for automated system startup. Examine and adapt each rule to site requirements. Change the CONTROL-M procedure name specified in the rule (CONTROLM) to the actual name of the CONTROL-M procedure at the site.

Table 13 Start CONTROL-D Monitor and Subsystem Rule Structure (part 1 of 2)

Item	Description
Title	Start CONTROL-D Monitor and Subsystem
Name	SCTD
Table	STARTSYS
Event	SCTD
Event Description	This event rule starts CONTROL-D under JES2.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.

Table 13 Start CONTROL-D Monitor and Subsystem Rule Structure (part 2 of 2)

Item	Description
Rule Logic	The rule issues a START command to start the CONTROL-D monitor and subsystem. CONTROL-D remains in status STARTING until JES is up, at which point CONTROL-D starts executing.
Rule Actions	Issues the following commands: ■ `S CONTROLD` ■ `S IOAINIT, OPTIONS=D`
Activating the Rule	The rule is activated immediately, once it has been ordered.
Recommended Mode or Category	Activate the rule in LOG mode during the testing period. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of this rule requires some customization.
Customization	STARTSYS rules are provided as guidelines for automated system startup. Examine and adapt each rule to site requirements. Change the CONTROL-D procedure name specified in the rule (CONTROLD) to the actual name of the CONTROL-D procedure at the site.

Table 14 Order Non-IPL Rules Rule Structure (part 1 of 2)

Item	Description
Title	Order Non-IPL Rules
Name	ORDERALL
Table	STARTSYS
Event	ORDERALL
Event Description	This Event rule orders non-IPL rules.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	IN CTO-STRT-TSO-ISSUED STAT IN CTO-STRT-CICS-ISSUED STAT
Global Variables	None.
Rule Logic	This rule is triggered after CONTROL-O has started TSO and CICS. TSO and CICS startup are the last actions performed by IPL rules, at which point IPL rules can be replaced with the New Day (non-IPL) rules. This rule issues a command to rebuild CONTROL-O rules.
Rule Actions	Issues the following command F `%%CONTROLO,O=ALL,REBUILD`
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.

Table 14 Order Non-IPL Rules Rule Structure (part 2 of 2)

Item	Description
Recommended Mode or Category	Activate the rule in LOG mode during the testing period. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of this rule requires some customization.
Customization	STARTSYS rules are provided as guidelines for automated system startup. Examine and adapt each rule to site requirements. Prerequisite conditions referenced by this rule must be deleted at time of IPL (see "SolveWare Initialization" on page 31).

SHUTSYS

SolveWare subject SHUTSYS automates system shutdown. Started tasks and systems are requested to shut down and then monitored until they are completely down. Once all systems are down, JES2 is ordered to shut down. Finally, after JES2 is down, CONTROL-O brings itself down.

NOTE

Since an unintended system shutdown can have very serious consequences, it is imperative that all rules contained in SolveWare subject SHUTSYS be carefully reviewed and adapted to site requirements.

Solutions Provided

SolveWare subject SHUTSYS contains the System Shutdown solution. This solution contains sample rules for automated system shutdown.

System Shutdown

This solution contains sample rules for automated shutdown using CONTROL-O. The shutdown process is initiated by issuing operator command SHUTSYS from the master console. The operator is then requested to issue command SHUTSYS again for confirmation. System shutdown actually starts only if command SHUTSYS is issued twice during one minute.

When shutdown is started, active system components (started tasks, initiators, lines, and so on.) are stopped. Checks are performed cyclically to determine the status of the various system components. If all systems are down, CONTROL-O is restarted with SUB=MSTR (that is, under MVS rather than under JES2) and JES2 is stopped. CONTROL-O brings itself down when JES2 termination is completed.

SHUTSYS rules are designed to provide the level of flexibility and modularity required for meeting the various shutdown requirements of different sites. Different system components (such as CONTROL-M and CICS) are stopped and monitored by different rules. These rules all have a similar logic and structure, and with minimal adaptation they can be easily duplicated to handle other system components.

Dependencies among SHUTSYS rules are indicated in the diagram on the following page.

NOTE

All prerequisite conditions defined in this solution must be initialized (deleted) at time of IPL. This is especially important for SHUTSYS rules, because the unintended addition of prerequisite conditions may initiate the undesired shutdown of system components. For more details, see the section on initializing SolveWare in Chapter 2, "STARTSYS."

Rules

The System Shutdown solution includes the following rules:

- SHUTSYS Command–Issued Once
- SHUTSYS Command–Issued Twice
- Stop JES2 Devices
- Stop CONTROL-D
- Stop CONTROL-M
- Stop CICS
- Stop TSO
- TSO Stopped–Cancel Active Users
- TSO Stopped–Termination Option
- CICS Ended
- TSO Ended
- Stop VTAM
- Check Status of JES2 Devices
- Check Status of Initiators
- Check Status of Started Tasks
- Start New CONTROL-O Before JES2 Shutdown
- Stop JES2
- JES2 Termination Complete

Rules Structure

The following tables describe the structures of the System Shutdown solution rules.

Table 15 SHUTSYS Command—Issued Once Rule Structure

Item	Description
Title	SHUTSYS Command Issued Once
Name	SHUTSYS
Table	SHUTSYS
Command	SHUTSYS This command can be issued at the MVS master console only.
Command Description	This Command rule is triggered when command SHUTSYS is issued once at the MVS master console.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	The rule is triggered when command SHUTSYS is issued once. The operator is instructed to issue the SHUTSYS command again for confirmation to initiate the shutdown process. When command SHUTSYS is issued twice in one minute, this rule is not triggered again, because it is overridden (see the following section).
Rule Actions	■ Suppresses the command. ■ Sends a message to the operator console, asking the operator to confirm system shutdown by issuing command SHUTSYS again.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of this rule little or no customization.

Table 16 SHUTSYS Command—Issued Twice (part 1 of 2)

Item	Description
Title	SHUTSYS Command Issued Twice
Name	SHUTSYS
Table	SHUTSYS
Command	SHUTSYS This command can be issued at the master console only. Rule threshold is two occurrences of the command in one minute.
Command Description	This Command rule is triggered when command SHUTSYS is issued twice within one minute, at the MVS master console.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	The following are counters for each event that is triggered by this rule: ■ %%CHKDEVS_COUNT ■ %%CHKINIT_COUNT ■ %%CHKSTCS_COUNT
Rule Logic	The rule is triggered when command SHUTSYS is issued twice and stops all started system components (initiators, lines, started tasks, and so on.) By specifying a higher priority and a CONTINUE SEARCH N (No) value, this rule overrides the rule described in the preceding section.
Rule Actions	■ Suppresses the command. ■ Sets Global variable %%CHKLINES_COUNT to 0. ■ Sets Global variable %%CHKPRTS_COUNT to 0. ■ Sets Global variable %%CHKLGNS_COUNT to 0. ■ Sets Global variable %%CHKINIT_COUNT to 0. ■ Sets Global variable %%CHKSTCS_COUNT to 0. ■ Adds prerequisite condition or date CTO-SHUT-SYS-ISSUED STAT.
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.

Table 16 SHUTSYS Command—Issued Twice (part 2 of 2)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as examples and guidelines for automated system shutdown. Examine and adapt each rule to site requirements. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see "SolveWare Initialization" on page 31.

Table 17 Stop JES2 Devices (part 1 of 2)

Item	Description
Title	Stop JES2 Devices
Name	SHUTDEVS
Table	SHUTSYS
Event	SHUTDEVS
Event Description	This Event rule stops all started JES2 devices.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	The rule is activated by command rule SHUTSYS. For each JES2 device type (that is, PRTS, LNES, LGNS and RMTS), the rule issues a command to display the started devices, and for each started device issues an appropriate $P command. All initiators are stopped by issuing the command $PI.

Table 17 Stop JES2 Devices (part 2 of 2)

Item	Description
Rule Actions	■ Issues command $PI. ■ Issues command $D U,LNES,STARTED to obtain information on started JES2 lines. ■ For each started line (received as response message $HASP628), issues an appropriate $P command. ■ Issues command $D U,LGNS,STARTED to obtain information on started JES2 or VTAM interfaces. ■ For each started JES2 or VTAM interface (received as response message $HASP628), issues an appropriate $P command. ■ Issues command $D U,PRTS,STARTED to obtain information on started JES2 printers. ■ For each started printer (received as response message $HASP628), issues an appropriate $P command. ■ Issues command $D U,RMTS,STARTED to obtain information on started JES2 remote terminals. ■ For each started remote terminal (received as response message $HASP628), issues an appropriate $P command.
Activating the Rule	The rule is activated by prerequisite condition or date CTO-SHUT-SYS-ISSUED STAT, which is added by command rule SHUTSYS.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of this rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see "SolveWare Initialization" on page 31.

Table 18 Stop CONTROL-D (part 1 of 2)

Item	Description
Title	Stop CONTROL-D
Name	SHUTCTD
Table	SHUTSYS
Event	SHUTCTD
Event Description	This Event rule stops CONTROL-D and verifies its termination.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None
Rule Logic	The rule is activated by command rule SHUTSYS. It first checks if CONTROL-D is active. If so, the rule issues a command to stop CONTROL-D. After waiting a predefined period of time, the rule checks the status of CONTROL-D. If CONTROL-D is still active, the operator is notified and, optionally, a predefined cancel command is issued.
Rule Actions	■ Issues command `D J` and analyzes response message IEE105I to determine the status of CONTROL-D. ■ If CONTROL-D is active, it does the following: – Issues a STOP command to stop CONTROL-D. – Waits a predefined period of time. The rule is delayed by issuing a dummy (%%NULL) command in command-response mode after setting the required delay period to variable %%TIMEOUT. The default delay period is 240 seconds. This delay period can be adapted to site requirements (see Customization in this table). – Issues command `D J` again and analyzes response message IEE105I to determine the status of CONTROL-D. ■ If CONTROL-D is still active, notifies operator and optionally issues a cancel command (see Customization in this table).
Activating the Rule	The rule is activated by prerequisite condition or date CTO-SHUT-SYS-ISSUED STAT, which is added by command rule SHUTSYS.

Table 18 Stop CONTROL-D (part 2 of 2)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. For ease of customization, the values that may require adaptation have been defined in AutoEdit variables that are set in DO SET statements at the beginning of the rule. Adapt these variables, described below, to site requirements. (To define a rule to shut down a different system component, duplicate this rule by specifying option INSERT (I) for the rule in the Rule List screen and then adapting the duplicate rule accordingly. ■ %%STC_NAME Started task to be stopped. The default is CONTROLD. ■ %%NORMAL_SHUT Console command to stop the started task. For example: `P %%STC_NAME` ■ %%FORCE_SHUT Console command to cancel the started task if it is active after the delay period. For example: `C%%STC_NAME`. The default value for this variable is %%NULL, which indicates that no cancel command is to be issued. ■ %%WAIT_TIME Number of seconds to wait between issuing the stop command and checking the status of the started task. This value is used as the %%TIMEOUT value and reflects the started task's acceptable shutdown time duration. The default is 240 seconds. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see "SolveWare Initialization" on page 31.

Table 19 Stop CONTROL-M (part 1 of 3)

Item	Description
Title	Stop CONTROL-M
Name	SHUTCTM
Table	SHUTSYS
Event	SHUTCTM

Table 19 Stop CONTROL-M (part 2 of 3)

Item	Description
Event Description	This Event rule stops CONTROL-M and verifies its termination.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	IN CTO-SHUT-SYS-ISSUED STAT
Global Variables	None.
Rule Logic	The rule is activated by command rule SHUTSYS. It first checks if CONTROL-M is active. If so, the rule issues a command to stop CONTROL-M. After waiting a predefined period of time, the rule checks the status of CONTROL-M. If CONTROL-M is still active, the operator is notified and optionally a predefined cancel command is issued.
Rule Actions	■ Issues command `D J` and analyzes response message IEE105I to determine the status of CONTROL-M. ■ If CONTROL-M is active, it does the following: – Issues a STOP command to stop CONTROL-M. – Adds prerequisite condition or date CTO-SHUT-TSO-ISSUED STAT. – Waits a predefined period of time. The rule is delayed by issuing a dummy (%%NULL) command in command-response mode, after setting the required delay period to variable %%TIMEOUT. The default delay period is 60 seconds. This delay period can be adapted to site requirements (see Customization in this table). – Issues command `D J` again and analyzes response message IEE105I to determine the status of CONTROL-M. – If CONTROL-M is still active, notifies operator and optionally issues a cancel command (see Customization in this table).
Activating the Rule	The rule is activated by prerequisite condition or date CTO-SHUT-SYS-ISSUED STAT, which is added by command rule SHUTSYS.

Table 19 Stop CONTROL-M (part 3 of 3)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.

Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements.
	
	For ease of customization, the values that may require adaptation have been defined in AutoEdit variables that are set in DO SET statements at the beginning of the rule. Adapt these variables, described below, to site requirements. (To define a rule to shut down a different system component, duplicate this rule by specifying option INSERT (I) for the rule in the Rule List screen and then adapting the duplicate rule accordingly.)
	
	■ %%STC_NAME
	Started task to be stopped. The default is CONTROLM.
	■ %%NORMAL_SHUT
	Console command to stop the started task,
	For example:
	`P %%STC_NAME`
	■ %%FORCE_SHUT
	Console command to cancel the started task if it is active after the delay period.
	For example:
	`C %%STC_NAME`
	The default value for this variable is %%NULL, which indicates that no cancel command is to be issued.
	■ %%WAIT_TIME
	Number of seconds to wait between issuing the stop command and checking the status of the started task. This value is used as the %%TIMEOUT value and reflects the started task's acceptable shutdown time duration. The default is 60 seconds.
	
	Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see "SolveWare Initialization" on page 31.

Table 20 Stop CICS (part 1 of 2)

Item	Description
Title	Stop CICS
Name	SHUTCICS
Table	SHUTSYS
Event	SHUTCICS
Event Description	This Event rule stops CICS and verifies its termination.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	The rule is activated by command rule SHUTSYS. It first checks if CICS is active. If so, the rule issues a command to stop CICS. After waiting a predefined period of time, the rule checks the status of CICS. If CICS is still active, the operator is notified and optionally, a predefined cancel command is issued.
Rule Actions	Issues command `D J` and analyzes response message IEE105I to determine the status of CICS. If CICS is active, it does the following: ■ issues a command to stop CICS ■ waits a predefined period of time The rule is delayed by issuing a dummy (%%NULL) command in command-response mode, after setting the required delay period to variable %%TIMEOUT. The default period is 300 seconds. This delay period can be adapted to site requirements (see Customization in this table). ■ issues command `D J` again and analyzes response message IEE105I to determine the status of CICS ■ if CICS is still active, notifies operator and optionally issues a cancel command (see Customization in this table)
Activating the Rule	The rule is activated by prerequisite condition or date CTO-SHUT-SYS-ISSUED STAT, which is added by command rule SHUTSYS.

Table 20 Stop CICS (part 2 of 2)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. For ease of customization, the values that may require adaptation have been defined in AutoEdit variables that are set in DO SET statements at the beginning of the rule. Adapt these variables, described below, to site requirements. (To define a rule to shut down a different system component, duplicate this rule by specifying option INSERT (I) for the rule in the Rule List screen and then adapting the duplicate rule accordingly.) ■ %%STC_NAME Started task to be stopped. The default is CICS. ■ %%NORMAL_SHUT Console command to stop the started task. For example: `F %%STC_NAME,CEMT P SHUT` ■ %%FORCE_SHUT Console command to cancel the started task if it is active after the delay period. The default value for this variable is %%NULL, which indicates that no cancel command is to be issued ■ %%WAIT_TIME Number of seconds to wait between issuing the stop command and checking the status of the started task. This value is used as the %%TIMEOUT value and reflects the started task's acceptable shutdown time duration. The default is 300 seconds. Prerequisite conditions referenced by this rule must be deleted at time of IPL.For more details, see "SolveWare Initialization" on page 31. If a standard facility for sending notification to CICS users is used at the site, an appropriate message can be sent to all CICS users notifying them of the shutdown. In this case, modify the rule to include a delay after CICS user notification to allow users time to save their data before CICS is stopped. For an example of such a delay, see the following section.

Table 21 Stop TSO (part 1 of 2)

Item	Description
Title	Stop TSO
Name	SHUTTSO
Table	SHUTSYS
Event	SHUTTSO
Event Description	This Event rule notifies TSO users of the shutdown, stops TSO and verifies its termination.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	The rule is activated by command rule SHUTSYS. It first checks if TSO is active. If so, the rule notifies all TSO users that shutdown is about to occur. After waiting a predefined period of time, the rule issues a command to stop TSO. After another delay, the rule checks the status of TSO. If TSO is still active, the operator is notified and optionally, a predefined cancel command is issued.
Rule Actions	Issues command D J and analyzes response message IEE105I to determine the status of TSO. If TSO is active, it does the following: ■ notifies all TSO users that system shutdown is about to occur ■ waits a predefined period of time and then issues a STOP command to stop TSO ■ waits a predefined period of time The rule is delayed by issuing a dummy (%%NULL) command in command-response mode, after setting the required delay period to variable %%TIMEOUT. The default delay period is 120 seconds. This delay period can be adapted to site requirements (see Customization in this table). ■ issues command D J again and analyzes response message IEE105I to determine the status of TSO ■ if TSO is still active, notifies operator and optionally issues a cancel command (see Customization in this table)
Activating the Rule	The rule is activated by prerequisite condition or date CTO-SHUT-SYS-ISSUED STAT, which is added by command rule SHUTSYS.

Table 21 Stop TSO (part 2 of 2)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. For ease of customization, the values that may require adaptation have been defined in AutoEdit variables that are set in DO SET statements at the beginning of the rule. Adapt these variables, described below, to site requirements. (To define a rule to shut down a different system component, duplicate this rule by specifying option INSERT (I) for the rule in the Rule List screen and then adapting the duplicate rule accordingly.) ■ %%STC_NAME Started task to be stopped. The default is TSO. ■ %%NORMAL_SHUT Console command to stop the started task. For example: P TSO ■ %%FORCE_SHUT Console command to cancel the started task if it is active after the delay period. The default value for this variable is %%NULL, which indicates that no cancel command is to be issued. ■ %%WAIT_TIME Number of seconds to wait between issuing the stop command and checking the status of the started task. This value is used as the %%TIMEOUT value and reflects the started task's acceptable shutdown time duration. The default is 120 seconds. ■ %%LOGOFF_TIME Number of seconds to wait between notifying TSO users and stopping TSO. The default is a delay of three minutes. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see "SolveWare Initialization" on page 31.

Table 22 TSO Stopped—Cancel Active Users

Item	Description
Title	TSO Stopped—Cancel Active Users
Name	IKT010D
Table	SHUTSYS
Message	`IKT010D nnnnn USERS ACTIVE, REPLY 'SIC' OR 'FSTOP'`
Message Description	TSO is terminating, but number of users nnnnn are still logged on.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	This rule is triggered during TSO shutdown. Message IKT010D is issued requesting instructions on how to handle active TSO logons. The rule automatically replies to the message with FSTOP.
Rule Actions	Replies to the message with FSTOP.
Activating the Rule	The rule is activated by prerequisite condition or date CTO-SHUT-TSO-ISSUED STAT, which is added by rule SHUTTSO (see Table 21 on page 57).
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. SIC and FSTOP in the message reply string can be changed according to site shutdown conventions. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see "SolveWare Initialization" on page 31.

Table 23 TSO Stopped—Termination Option

Item	Description
Title	TSO Stopped—Termination Option
Name	IKT012D
Table	SHUTSYS
Message	`IKT012D TCAS TERMINATION IN PROGRESS – SPECIFY 'U' OR 'DUMP'`
Message Description	TSO is terminating. An SVC dump can optionally be requested.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	This rule is triggered during TSO shutdown. Message IKT012D is issued asking whether a dump is required. The rule automatically replies to the message with U to terminate without a dump.
Rule Actions	Replies to the message with U.
Activating the Rule	The rule is activated by prerequisite condition or date CTO-SHUT-TSO-ISSUED STAT, which is added by rule SHUTTSO (see Table 21 on page 57).
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see "SolveWare Initialization" on page 31.

Table 24 CICS Ended (part 1 of 2)

Item	Description
Title	CICS Ended
Name	IEF404I
Table	SHUTSYS
Message	`IEF404I` *`jjj`* `ENDED{ - TIME=`*`hh.mm.ss`*`}` when the message is issued for job or STC CICS
Message Description	Job or STC *jjj* ended.
Basic Scheduling Parameters	Always schedule this rule.

Table 24 CICS Ended (part 2 of 2)

Item	Description
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	This rule is activated by command rule SHUTSYS and is triggered when CICS is fully stopped. The rule handles the dependency between CICS termination and VTAM shutdown. Upon CICS termination, the rule adds prerequisite condition or date CTO-SHUT-CICS-DOWN STAT. This condition is defined as an IN condition for rule SHUTVTAM.
Rule Actions	Adds prerequisite condition or date CTO-SHUT-CICS-DOWN STAT.
Activating the Rule	The rule is activated by prerequisite condition or date CTO-SHUT-SYS-ISSUED STAT, which is added by command rule SHUTSYS.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see "SolveWare Initialization" on page 31.

Table 25 TSO Ended (part 1 of 2)

Item	Description
Title	TSO Ended
Name	IEF404I
Table	SHUTSYS
Message	`IEF404I jjj ENDED{ - TIME=hh.mm.ss}` when the message is issued for STC TSO
Message Description	STC *jjj* ended.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.

Table 25 TSO Ended (part 2 of 2)

Item	Description
Rule Logic	This rule is activated by command rule SHUTSYS and is triggered when TSO is fully stopped. The rule handles the dependency between TSO termination and VTAM shutdown. Upon TSO termination, the rule adds prerequisite condition or date CTO-SHUT-TSO-DOWN STAT. This condition is defined as an IN condition for rule SHUTVTAM.
Rule Actions	Adds prerequisite condition or date CTO-SHUT-TSO-DOWN STAT.
Activating the Rule	The rule is activated by prerequisite condition or date CTO-SHUT-SYS-ISSUED STAT, which is added by command rule SHUTSYS.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see “SolveWare Initialization” on page 31.

Table 26 Stop VTAM (part 1 of 3)

Item	Description
Title	Stop VTAM
Name	SHUTVTAM
Table	SHUTSYS
Event	SHUTVTAM
Event Description	This Event rule stops VTAM and verifies its termination.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	The rule is activated by command rule SHUTSYS. It first checks if VTAM is active. If so, the rule issues a command to stop VTAM. After waiting a predefined period of time, the rule checks the status of VTAM. If VTAM is still active, the operator is notified and optionally a predefined cancel command is issued.

Table 26 Stop VTAM (part 2 of 3)

Item	Description
Rule Actions	Issues command `D J` and analyzes response message IEE105I to determine the status of VTAM. If VTAM is active, it does the following: ■ issues command `Z NET,QUICK` to stop VTAM ■ waits a predefined period of time The rule is delayed by issuing a dummy (%%NULL) command in command-response mode, after setting the required delay period to variable %%TIMEOUT. The default delay period is 180 seconds. This delay period can be adapted to site requirements (see Customization in this table). ■ issues command `D J` again and analyzes response message IEE105I to determine the status of VTAM ■ if VTAM is still active, notifies operator and optionally issues a cancel command (see Customization in this table)
Activating the Rule	The rule is activated when both prerequisite condition or date CTO-SHUT-TSO-DOWN STAT and CTO-SHUT-CICS-DOWN STAT have been added upon full termination of TSO and CICS (respectively).

Table 26 Stop VTAM (part 3 of 3)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. For ease of customization, the values that may require adaptation have been defined in AutoEdit variables that are set in DO SET statements at the beginning of the rule. Adapt these variables, described below, to site requirements. (To define a rule to shut down a different system component, duplicate this rule by specifying option INSERT (I) for the rule in the Rule List screen and then adapting the duplicate rule accordingly.) ■ ■ %%STC_NAME Started task to be stopped. The default is VTAM. ■ %%NORMAL_SHUT Console command to stop the started task. For example: Z NET,QUICK ■ %%FORCE_SHUT Console command to cancel the started task if it is active after the delay period. The default is Z NET, CANCEL. This variable can contain %%NULL to indicate that no cancel command is to be issued. ■ %%WAIT_TIME Number of seconds to wait between issuing the stop command and checking the status of the started task. This value is used as the %%TIMEOUT value and reflects the started task's acceptable shutdown time duration. The default is 180 seconds. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see "SolveWare Initialization" on page 31.

Table 27 Check Status of JES2 Devices (part 1 of 2)

Item	Description
Title	Check Status of JES2 Devices
Name	CHKDEVS
Table	SHUTSYS
Event	CHKDEVS
Event Description	This Event rule checks the status of JES2 devices during shutdown.

Table 27 Check Status of JES2 Devices (part 2 of 2)

Item	Description
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	IN CTO-SHUT-SYS-ISSUE STAT IN !CTO-SHUT-DEVS-DOWN STAT INTERVAL 001
Global Variables	%%CHKDEVS_COUNT Counter for cyclic triggering.
Rule Logic	The rule is activated by command rule SHUTSYS. It issues a command to display started JES2 devices and waits for a response. If no started devices are found (indicated by response message $HASP668 NO DEVICES FOUND), the rule adds prerequisite condition or date CTO-SHUT-DEVS-DOWN STAT. If not all devices have stopped after a predefined period of time, a message is sent to the operator console and the rule continues to monitor device status.
Rule Actions	■ Sets variable %%$WAITRESP to YES. ■ Sets variable %%$RESPMSG to $HASP668. ■ Issues the following command: `$D U,STARTED` ■ If response message $HASP668 was not received, increases Global variable %%CHKDEVS_COUNT by 1. If %%CHKDEVS_COUNT is equal to 5, sets it to 0 and notifies console operator. ■ If response message $HASP668 was received, adds prerequisite condition or date CTO-SHUT-DEVS-DOWN STAT.
Activating the Rule	The rule is activated by prerequisite condition or date CTO-SHUT-SYS-ISSUED STAT, which is added by command rule SHUTSYS.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see “SolveWare Initialization” on page 31.

Table 28 Check Status of Initiators (part 1 of 2)

Item	Description
Title	Check Status of Initiators
Name	CHKINIT
Table	SHUTSYS
Event	CHKINIT
Event Description	This Event rule checks the status of JES2 initiators (JES2 or VTAM interfaces) during shutdown.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	%%CHKINIT_COUNT Counter for cyclic triggering.
Rule Logic	The rule is activated by command rule SHUTSYS. It issues a command to display started initiators and waits for a response. The number of started initiators is obtained from response message IEE104I. If no started initiators are found, the rule adds prerequisite condition or date CTO-SHUT-INIT-DOWN STAT. If not all initiators have stopped after a predefined period of time, a message is sent to the operator console and the rule continues to monitor initiator status.
Rule Actions	■ Sets variable %%$WAITRESP to YES. ■ Sets variable %%$RESPMSG to IEE104I. ■ Issues the following command: `D A` ■ Determines the number of started initiators according to response message IEE104I. ■ If the number of started initiators is not 0, increases Global variable %%CHKINIT_COUNT by 1. If %%CHKINIT_COUNT is equal to 5, sets it to 0 and notifies console operator. ■ If the number of started initiators is 0, adds prerequisite condition or date CTO-SHUT-INIT-DOWN STAT.
Activating the Rule	The rule is activated by adding prerequisite condition or date CTO-SHUT-SYS-ISSUED STAT, which is added by command rule SHUTSYS.

Table 28 Check Status of Initiators (part 2 of 2)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see "SolveWare Initialization" on page 31.

Table 29 Check Status of Started Tasks (part 1 of 2)

Item	Description
Title	Check Status of Started Tasks
Name	CHKSTCS
Table	SHUTSYS
Event	CHKSTCS
Event Description	This Event rule checks the status of started tasks during shutdown.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	%%CHKSTCS_COUNT Counter for cyclic triggering.
Rule Logic	The rule is activated by command rule SHUTSYS. It issues a command to display started tasks and waits for a response. The number of started tasks is obtained from response message IEE104I. If the number of started tasks is two (that is, CONTROL-O and JES2), the rule adds a prerequisite condition. This default value for the maximum number of started tasks may require customization (see Customization in this table). If any of the started tasks has not stopped after five minutes, a message is sent to the operator console and the rule continues to monitor started task status.

Table 29 Check Status of Started Tasks (part 2 of 2)

Item	Description
Rule Actions	■ Sets variable %%$WAITRESP to YES. ■ Sets variable %%$RESPMSG to IEE104I. ■ Issues the following command: `D A` ■ Determines the number of started tasks according to response message IEE104I. ■ If the number of started tasks is not two, increases Global variable %%CHKSTCS_COUNT by one. If %%CHKSTCS_COUNT is equal to 5, sets it to 0 and notifies console operator. ■ If the number of started tasks is two, deletes prerequisite condition or date CTO-SHUT-SYS-ISSUED STAT and adds prerequisite condition or date CTO-SHUT-STCS-DOWN STAT.
Activating the Rule	The rule is activated by prerequisite condition or date CTO-SHUT-SYS-ISSUED STAT, which is added by command rule SHUTSYS.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. The rule specifies two as the maximum number of started tasks to ignore (that is, JES2 and CONTROL-O), but this value may need adapting to site requirements. For example, since the Functional SubSystem (FSS) is stopped by JES2 during JES2 termination, ignore it when checking the status of started tasks. In this case, increase the number of ignored started tasks by one. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see “SolveWare Initialization” on page 31.

Table 30 Start New CONTROL-O Before JES2 Shutdown (part 1 of 2)

Item	Description
Title	Start New CONTROL-O Before JES2 Shutdown
Name	NEWCTO
Table	SHUTSYS
Event	NEWCTO

Table 30 Start New CONTROL-O Before JES2 Shutdown (part 2 of 2)

Item	Description
Event Description	This Event rule starts a new CONTROL-O monitor to run under the MVS master subsystem.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	This rule is triggered after all system components have stopped. CONTROL-O is started under the MVS master subsystem (that is, with SUB=MSTR) in order to enable JES2 termination.
Rule Actions	Issues the following command `S %%CONTROLO,SUB=MSTR, OUTPUT=DUMMY`
Activating the Rule	The rule is activated when all system components (apart from JES2 tasks and CONTROL-O) are stopped. The appropriate prerequisite conditions are added by several rules (described in this solution) that monitor shutdown operations.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see "SolveWare Initialization" on page 31.

Table 31 Stop JES2 (part 1 of 2)

Item	Description
Title	Stop JES2
Name	$HASP099
Table	SHUTSYS
Message	`$HASP099 ALL AVAILABLE FUNCTIONS COMPLETE`
Message Description	JES2 has no more work to do. All job processors are dormant and all RJE lines are inactive.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.

Table 31 Stop JES2 (part 2 of 2)

Item	Description
Rule Logic	This rule is activated by rule NEWCTO (rule NEWCTO starts a new CONTROL-O monitor, which is required before JES2 can be stopped) and is triggered when JES2 has no more work to perform. At this point JES2 is ready to be stopped and the rule issues command $PJES2 to stop JES2.
Rule Actions	Issues the following command: `$PJES2` Adds prerequisite condition or date CTO-SHUT-JES2-DOWN STAT
Activating the Rule	The rule is activated by prerequisite condition or date CTO-SHUT-CTO-ISSUED STAT, which is added by rule NEWCTO.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see "SolveWare Initialization" on page 31.

Table 32 JES2 Termination Complete (part 1 of 2)

Item	Description
Title	JES2 Termination Complete
Name	$HASP085
Table	SHUTSYS
Message	`$HASP085 JES2 TERMINATION COMPLETE`
Message Description	JES2 has completed a requested shutdown.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	After JES2 termination, this rule notifies operator, issues command HALT EOD and stops CONTROL-O.

Table 32 JES2 Termination Complete (part 2 of 2)

Item	Description
Rule Actions	Deletes prerequisite condition or date CTO-SHUT-JES2-ISSUED STAT Issues the following command: `HALT EOD` Notifies console operator that shutdown is complete. Issues the following command: `F %%CONTROLO,STOP`
Activating the Rule	The rule is activated by prerequisite condition or date CTO-SHUT-SYS-ISSUED STAT, which is added by rule SHUTSYS.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.
Customization	SHUTSYS rules are provided as guidelines for automated system shutdown. Examine and adapt each rule to site requirements. Prerequisite conditions referenced by this rule must be deleted at time of IPL. For more details, see “SolveWare Initialization” on page 31.

Chapter 4

SUPPRESS

The following solutions suppress system console messages that are not required by system operators. Suppressing such messages enables operators to receive only relevant messages that can then be investigated and handled.

NOTE

The CONTROL-O Message Statistics screen provides message statistics that are useful in determining which messages to suppress. Before deciding to suppress a message, read the detailed message description in the message guide of the relevant system component.

Individual messages are suppressed by individual rules. This means that each message is suppressed by the rule that contains its message ID.

NOTE

The suppression rules for all messages in any of the following solutions are identical except for the message ID. Therefore, only the rule for the first listed message in each solution is detailed as a sample.

Solutions Provided

SolveWare subject SUPPRESS contains the following solutions:

- MVS Message Suppression – Suppresses certain MVS messages that are not required by operators.
- JES2 Message Suppression – Suppresses certain JES2 messages that are not required by operators.
- JES3 Message Suppression – Suppresses certain JES3 messages that are not required by operators.

- System Component Message Suppression – Suppresses various system component messages that are not required by operators.

MVS Message Suppression

This solution contains rules to suppress certain MVS messages from the system console. These messages are not suppressed from the system log.

Each message is suppressed by the rule that contains its message ID.

Rules

Table 33 lists the rules and the MVS messages that they suppress.

Table 33 MVS Message Suppression Rules (part 1 of 2)

Rule	Message
IEA848I	IEA848I NO DUMP WAS PRODUCED FOR THIS ABEND, DUE TO SYSTEM OR INSTALLATION REQUEST
IEA989I	IEA989I SLIP TRAP ID = xxxx MATCHED
IEA995I	IEA995I SYMPTOM DUMP OUTPUT ABEND CODE {SYSTEM | USER} = code TIME=hh.mm.ss SEQ=number CPU=n ASID=asn PSW AT TIME OF ERROR pppppppp pppppppp ILC x INTC y ACTIVE LOAD MODULE = modname ADDRESS = aaaaaaaa OFFSET = dddd DATA AT PSW -xxxxxxxx - xxxxxxxx xxxxxxxx xxxxxxxx GPR 0-3 R00 R01 R02 R03 GPR 4-6 R04 R05 R06 R07 GPR 8-11 R08 R09 R10 R11 GPR 12-15 R12 R13 R14 R15 END OF SYMPTOM DUMP
IEC070I	IEC070I rc{(sfi)}-ccc,jjj,sss,ddn,ddd,ser,xxx,dsn,cat
IEC130I	IEC130I ddn - DD STATEMENT MISSING
IEC141I	IEC141I 013-rc,mod,jjj,sss,ddn{-#},ddd,ser,DSN
IEC161I	IEC161I rc{(sfi)}-ccc,jjj,sss,ddn,ddd,ser,xxx,dsn,cat
IEC331I	IEC331I rc-crs,jjj,sss,proc,mmm
IEC705I	IEC705I TAPE ON ddd,ser {IS}{,labtyp}{trtch},den BPI{,jjj,sss}{,dsn}

Table 33 MVS Message Suppression Rules (part 2 of 2)

Rule	Message
IEC999I	IEC999I mod,sub,debaddr = xxxxxx
	or
	IEC999I mod,sub,workarea = xxxxxx
	or
	IEC999I mod,jjj,sss{,ddd,ser,dsn}
IEE043I	IEE043I A SYSTEM LOG DATASET HAS BEEN QUEUED TO SYSOUT CLASS x
IEE400I	IEE400I THESE MESSAGES CANCELED - nn,nn,nn,...
IEF097I	IEF097I USER userid AND GROUP groupid ASSIGNED
IEF097I	IEF097I jobname USER userid ASSIGNED
IEF125I	IEF125I jjj- LOGGED ON {- TIME hh:mm:ss}
IEF126I	IEF126I jjj- LOGGED OFF {- TIME hh:mm:ss}
IEF170I	IEF170I n jjj (msg)
IEF188I	IEF188I PROBLEM PROGRAM ATTRIBUTES ASSIGNED
IEF196I	IEF196I jcl-text
IEF202I	IEF202I STEP sss WAS NOT RUN BECAUSE OF cde
IEF236I	IEF236I ALLOC. FOR jjj {ppp} sss
IEF237I	IEF237I ddd ALLOCATED TO ddn
IEF287I	IEF287I dsn dsp w VOL SER NOS = ser, ser, ser, ser, ser, ser
IEF288I	IEF288I dsn SYSOUT
IEF403I	IEF403I jjj STARTED{ - TIME=hh.mm.ss}
IEF404I	IEF404I jjj ENDED{ - TIME=hh.mm.ss}
IEF452I	IEF452I jjj - JOB NOT RUN - JCL ERROR
IEF452I	IEF452I ppp - JOB NOT RUN - JCL ERROR
IEF452I	IEF452I JOBFAIL - JOB NOT RUN - JCL ERROR
IEF453I	IEF453I jjj - JOB NOT RUN - JCL ERROR
IEF677I	IEF677I WARNING MESSAGES FOR JOB jjj ISSUED
IEF722I	IEF722I jjj - FAILED txt

Rules Structure

The following table details the structure of the MVS message suppression rules. All MVS messages included in this solution are identical, except for their message IDs.

Table 34 MVS Message Suppression Rule Structure

Item	Description
Title	MVS Message Suppression
Name	*message-Id* *message-id* can be any name (message ID), as listed in Table 33.
Table	SUPMVS
Message	*message-id* NO DUMP WAS PRODUCED FOR THIS ABEND, DUE TO SYSTEM OR INSTALLATION REQUEST
Message Description	No dump was produced for a certain abend situation.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	The rule suppresses the message from the operator console. The message still appears in the system log.
Rule Actions	Suppresses the message from the system console.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	Activate the rule in PROD mode. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If suppression of the particular message is not required, delete the rule from the table.

JES2 Message Suppression

This solution contains rules to suppress certain JES2 messages from the system console. These messages are not suppressed from the system log.

Each message is suppressed by the rule that contains its message ID.

Rules

Table 35 lists the rules and the JES2 messages that they suppress.

Table 35 JES2 Message Suppression Rules

Rule	Message
$HASP100	$HASP100 jobname ON device {progname} {FROM ijobid ijobname}
$HASP101	$HASP101 jobname HELD
$HASP110	$HASP110 jobname {ILLEGAL JOB CARD | MULTIPLE JOB CARDS}
$HASP111	$HASP111 jobname – INVALID /*ROUTE CARD
$HASP112	$HASP112 jobname – INVALID /*JOBPARM CARD
$HASP113	$HASP113 jobname – INVALID /*OUTPUT CARD
$HASP114	$HASP114 jobname – INVALID EXECUTION NODE
$HASP115	$HASP115 jobname – INVALID /*NETACCT CARD
$HASP116	$HASP116 jobname – INVALID /*NOTIFY CARD
$HASP117	$HASP117 jobname – INVALID /*XMIT CARD
$HASP118	$HASP118 jobname – INVALID /* CONTROL STATEMENT
$HASP125	$HASP125 jobname SKIPPING FOR JOB CARD
$HASP150	$HASP150 jobname OUTGRP=groupid.jobid1.jobid2 ON device {nnnnnnnn (mmmmmmmm) LINES | nnnnnnnn (mmmmmmmm) RECORDS | nnnnnnnn (mmmmmmmm) PAGES}
$HASP160	$HASP160 device INACTIVE– CLASS = class1 class2 ...
$HASP250	$HASP250 jobname IS PURGED
$HASP301	$HASP301 jobname – DUPLICATE JOBNAME – JOB DELAYED
$HASP309	$HASP309 {XB | bb} INIT nn INACTIVE {******** |monitor} C = classes
$HASP373	$HASP373 jobname STARTED{– INIT nnn – CLASS x – SYS ssss}
$HASP395	$HASP395 jobname ENDED
$HASP523	$HASP523 jobname-device DELETED
$HASP524	$HASP524 devname INACTIVE
$HASP530	$HASP530 jobname ON device {nnnnn RECORDS}
$HASP532	$HASP532 jobname-device RESTARTED{, JOB HELD}
$HASP533	$HASP533 jobname-device DELETED
$HASP534	$HASP534 devname INACTIVE
$HASP540	$HASP540 jobname ON device {nnnnn RECORDS}

Rules Structure

The following table details the structure of the JES2 message suppression rules. All JES2 messages included in this solution are identical, except for their message IDs.

Table 36 JES Message Suppression Rule Structure

Item	Description
Title	JES2 Message Suppression
Name	*message-id* *message-id* can be any name (message ID), as listed in Table 35.
Table	SUPJES2
Message	*message-id* jobname ON device {progname} {FROM ijobid ijobname}
Message Description	JES2 received a JOB statement from device.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	The rule suppresses the messages from the operator console. The message still appears in the system log.
Rule Actions	Suppresses the message from the system console.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	Activate the rule in PROD mode. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If suppression of the particular message is not required, delete the rule from the table.

JES3 Message Suppression

This solution contains rules to suppress certain JES3 messages from the system console. These messages are not suppressed from the system log.

Each message is suppressed by the rule that contains its message ID.

Rules

Table 37 lists the rules and the JES3 messages that they suppress.

Table 37 JES3 Message Suppression Rules

Rule	Message
IAT2000	IAT2000 JOB jobname(jobid) SELECTED main GRP=group {NET-ID=djnet}
IAT2006	IAT2006 PREMATURE JOB TERM – JOB jobname(jobid) {HELD | RESTARTED | PRINTED | CANCELED}
IAT5200	IAT5200 JOB jobname (jobid) IN SETUP ON MAIN = main
IAT6101	IAT6101 {(jobid) | (DEMSEL) | JOB {jobno | jobname}} jobid2 IS jobname PRTY=nnn [HOLD] [NETID=djnet]
IAT6108	IAT6108 JOB jobname (jobid) text [,NOTIFY NOT SENT, NO PATH]
IAT6118	IAT6118 nn CARDS FLUSHED [BEFORE JOB CARD FOUND]
IAT7001	IAT7001 JOB jobname (jobid) IS ON WRITER devname (ddd), RECORDS = {nnn | NAVAIL} {PAGES = mmm}{(RESTARTED)}
IAT7007	IAT7007 JOB jobname (jobid) ON WRITER devname (ddd),DD=dsn, PURGED.
IAT7120	IAT7120 I/O ERROR ON con STATUS stat SENSE sens OP nn
IAT7450	IAT7450 JOB jobname (jobid) PURGED
IAT9123	IAT9123 DATA RECEPTION ACTIVE ON LINE lname
IAT9124	IAT9124 JOB RECEPTION ACTIVE ON LINE lname
IAT9127	IAT9127 JOB (jobid) IS jobname FROM nodename (userid)
IAT9190	IAT9190 JOB jobname (jobid) {JB | OP} IS BEING SENT ON LINE lname
IAT9191	IAT9191 JOB jobname (jobid) {JB | OP} SENT TO NODE nodename ON LINE lname

Rules Structure

The following table details the structure of the JES3 message suppression rules. All JES2 messages included in this solution are identical, except for their message IDs.

Table 38 JES3 Message Suppression Rule Structure

Item	Description
Title	JES3 Message Suppression
Name	*message-id* *message-id* can be any name (message ID), as listed in Table 37.
Table	SUPJES3
Message	*message-id* JOB *jobname*(*jobid*) SELECTED *main* GRP=*group* {NET-ID=*djnet*}
Message Description	Job jobname is scheduled to execute on a certain main system and group.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	The rule suppresses the message from the operator console. The message still appears in the system log.
Rule Actions	Suppresses the message from the system console.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	Activate the rule in PROD mode. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If suppression of the particular message is not required, delete the rule from the table.

System Component Message Suppression

This solution contains rules to suppress certain TSO, VTAM, RACF, and DFHSM messages from the system console. These messages are not suppressed from the system log. Each message is suppressed by the rule that contains its message ID.

Rules

Table 39, Table 40, Table 41, and Table 42 list the rules and the TSO, VTAM, RACF, and DFHSM messages, respectively, that they suppress.

Table 39 TSO Message Suppression Rules

Rule	Message
IKJ144I	IKJ144I UNDEFINED USERID userid1 .. useridn
IKJ572I	IKJ572I USER userid NOT LOGGED ON, MESSAGE CANCELED
IKJ605I	IKJ605I TSOLOGON TERMINATED. TOO MANY ATTEMPTS. USER {userid | UNKNOWN}
IKJ606I	IKJ606I TSOLOGON REJECTED. USERID, userid, IN USE
IKT100I	IKT100I USERID userid CANCELED DUE TO UNCONDITIONAL LOGOFF.
IKT108I	IKT108I userid {SEND|RECEIVE} ERROR, RPLRTNCD=aa, RPLFDB2=bb, SENSE=code,WAITING FOR RECONNECTION termid.
IST532I	IST532I EVENT CODE = aaaaaaa EVENT ID = bbbbbbbb

Table 40 VTAM Message Suppression Rules

Rule	Message
IST234I	IST234I I/O ERROR termname, cmmod, ncpresp[, bstatus]
IST259I	IST259I INOP RECEIVED FOR nodename CODE = code [text]
IST530I	IST530I aaaaaaaa PENDING {FROM bbbbbbbb} {TO cccccccc } {FOR ddddddd}
IST532I	IST532I EVENT CODE = aaaaaaa EVENT ID = bbbbbbbb

Table 41 RACF Message Suppression Rules

Rule	Message
ICH408I	ICH408I USER(*userid*) GROUP(*group*) NAME(*username*) ICH408I {*resource-name*} {CL(*class-name*)} {VOL(*vol*)} ICH408I *variable text* ICH408I {FROM *generic-profile-name* (G)} ICH408I {ACCESS INTENT(*intent*) ACCESS ALLOWED(*allowed*)}
	or
	ICH408I JOB(*jobname*) STEP(*stepname*) SUBMITTER(*userid*) ICH408I {*resource-name*} {CL(*class-name*)} {VOL(*vol*)} ICH408I *variable text* ICH408I {FROM *generic-profile-name* (G)} ICH408I {ACCESS INTENT(*intent*) ACCESS ALLOWED(*allowed*)}
ICH70001I	ICH70001I *userid* LAST ACCESS AT *hh:mm:ss* ON *www,month,day,yyyy*

Table 42 DFHSM Message Suppression Rules (part 1 of 2)

Rule Name	Message Suppressed
ARC0100I	ARC0100I {RELEASE | HOLD | SETSYS} COMMAND COMPLETED
ARC0200I	ARC0200I TRAP IN MODULE pgm, CODE=cde, {LOG ALWAYS ADDED | NODUMP ONCE ADDED | FATAL ONCE ADDED | DEBUG ALWAYS ADDED | SNAP {ALWAYS | ONCE | NEVER} ADDED | ABEND {ALWAYS | ONCE | NEVER} ADDED | REMOVED}
ARC0208I	ARC0208I TRAP FOR ERROR CODE errcode, MODULE modname,{NODUMP ONCE | FATAL ONCE | DEBUG ALWAYS | 200 MAXLIMIT | SNAP {ALWAYS | ONCE | NEVER} | ABEND{ALWAYS | ONCE | NEVER}
ARC0503I	ARC0503I ALLOCATION ERROR, {VOLUME=volser | DATASET =dsname | DD DUMMY}, RETURN CODE=rc, REASON CODE=reascode, INFO CODE=infocode

Table 42 DFHSM Message Suppression Rules (part 2 of 2)

ARC0728I	`ARC0728I VTOC FOR VOLUME volser1 COPIED TO DATASET dsname ON volser2`
ARC0734I	`ARC0734I ACTION=action FRVOL=volser1 TOVOL=volser2 {BLOCKS=blocks|TRACKS=tracks} RC=retcde, REASON=rsncde, AGE=days, DSN=dsname`

Rules Structure

The following table details the structure of the TAO, VTAM, RACF, and DFHSM message suppression rules. All JES2 messages included in this solution are identical, except for their message IDs.

Table 43 System Component Message Suppression Rule Structure

Item	Description
Title	System Component Message Suppression
Name	*message-id* *message-id* can be any name (message ID), as listed in Table 39 for TSO messages, Table 40 for VTAM messages, Table 41 for RACF messages, and Table 42 for DFHSM messages.
Table	SUPSYS
Message	*`message-id`* `UNDEFINED USERID` *`userid1 .. useridn`*
Message Description	An operator SEND command contains an undefined user. This user is ignored.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	The rule suppresses the message from the operator console. The message still appears in the system log.
Rule Actions	Suppresses the message from the system console.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	Activate the rule in PROD mode. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If suppression of the particular message is not required, delete the rule from the table.

DEVICES

The DEVICES solutions help maximize device utilization. They automate device-controlling actions that otherwise require manual response and they adjust IOA resources based on physical device monitoring.

Solutions Provided

SolveWare subject DEVICES contains the following solutions:

- Device Status Change – Recognizes situations in which a device is varied offline by an operator command and starts the deallocation procedure or any other started task (STC) to force the device offline.
- Device Quantitative Resource Handling – Updates the information held in the CONTROL-M Resources file according to events impacting device availability (for example: mount or demount requests, device online or offline).
- Device-Not-Available Handling, Option 1 – Handles device-not-available situations. A job requesting an unavailable device receives a WAIT reply followed by a NOHOLD reply.
- Device-Not-Available Handling, Option 2 – Similar to Device-Not-Available, Option 1, except that Option 2 also distinguishes between test and production jobs.
- Device-Not-Available Handling (in a Multisystem Environment), Option 3 – Checks other systems for a free device that can be varied online to the requesting system.

Device Status Change

Under MVS, when a device is varied offline by an operator command, the device is switched to status PENDING OFF-LINE. When allocation processing allocates any device in the system (for example, at the start of any job in the system), the PENDING OFF-LINE device then goes offline. This solution recognizes the situation in which a device is pending offline and starts the IBM-supplied DEALLOC dummy procedure to force the device offline immediately.

Rules

The Device Status Change solution includes the **Device Pending Offline** rule.

Rules Structure

The following tables describe the structure of the Device Status Change solution rules.

Table 44 Device Pending Offline Rule Structure (part 1 of 2)

Item	Description
Title	**Device Pending Offline**
Name	**IEE794I**
Table	**DEVICES**
Message	`IEE794I` *`ddd`* `PENDING OFFLINE`
Message Description	Device *ddd* is pending offline as a result of an operator command.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	When a device is pending offline, initiate the IBM supplied DEALLOC procedure, or any other dummy started task (that calls program IEFBR14), to force the device offline.
Rule Actions	Suppresses the message from the console. Starts the DEALLOC started task (STC) to deallocate the device.

Table 44 Device Pending Offline Rule Structure (part 2 of 2)

Item	Description
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is 3—this rule is provided as an example. Implementation of the rule requires little or no customization.

Device Quantitative Resource Handling

Quantitative resources are commonly used to control device availability. For example, if a certain job in CONTROL-M requires two tape drives, CONTROL-M will not submit the job unless two drives are available.

Normally, Quantitative resources mechanism determines resource availability based only on values defined within the IOA environment. Devices manually brought online or offline or devices used by a manually submitted job (that is, a job not submitted by CONTROL-M) are not reflected in IOA resources.

This solution's rules monitor external changes to Quantitative resources by detecting device online or offline and device mount or demount messages and then adjust (increase or decrease) the resource quantity accordingly.

Global variables are used to define device-related information: device name, type, status, counters, name of the related Quantitative resource and (in a multi-system configuration) the CPU configuration used. These Global variables are defined during the SolveWare initialization by rule JES2GLBL in table INITSLV. For details, see the section on initializing SolveWare Chapter 2, "STARTSYS."

Rules

The Device Quantitative Resource Handling solution includes the following rules:

- Initialize Device Global Variables
- Device Brought Online
- Device Brought Offline
- Device Mount Request
- Device Demount Request
- Job Ended – Release Device Quantitative Resources
- Device Swap in Progress

Rules Structure

The following tables describe the structures of the Device Quantitative Resource Handling solution rules.

Table 45 Initialize Device Global Variables Rule Structure (part 1 of 4)

Item	Description
Title	**Initialize Device Global Variables**
Name	**DEVTYPES**
Table	**DEVICES**
Event	DEVTYPES
Event Description	This Event rule sets the values of Global variables referenced by other rules in the same solution.
Basic Scheduling Parameters	Schedule this rule if initialization or update of device-type Global variables is desired.
Runtime Scheduling Parameters	PRIORITY 20

Table 45 Initialize Device Global Variables Rule Structure (part 2 of 4)

Item	Description

Global Variables	■ %%DTYPE_TIME
	Last time the rule was executed.
	■ %%DEVNAME_*type*
	Generic name of the device controlled by CONTROL-O, as defined to MVS.
	For example:
	`%%DEVNAME_3400=TAPE`
	`%%DEVNAME_348S=CART`
	■ %%RESNAME_*name*
	Name of the Quantitative resource used by CONTROL-M for the device it controls.
	For example:
	`%%RESNAME_TAPE=TAPE %%RESNAME_CART=CARTRIDGE`
	■ %%DEV_*index*
	Generic device name, indexed by a sequence number.
	For example:
	`%%DEV_1=TAPE, %%DEV_2=CART`
	■ %%DEV_TOTAL
	Total number of all devices units types controlled by CONTROL-O.
	For example:
	`%%DEV_TOTAL=20`
	■ %%DEV_INDEX_*index*
	Device unit address, indexed by a sequence number.
	For example:
	`%%DEV_INDEX_1=460`
	`%%DEV_INDEX_2=461`

Table 45 Initialize Device Global Variables Rule Structure (part 3 of 4)

Item	Description

	■ %%DEV_O_*type*
	Total number of devices of the specified *type* that are online.
	For example:
	`%%DEV_O_CART=0` if there are no online cartridges
	`%%DEV_O_TAPE=5` if there are five tapes online
	■ %%DEV_T_*type*
	Total number of devices of the specified *type*.
	For example:
	`%%DEV_T_CART=0` if no cartridge was defined in the installation
	`%%DEV_T_TAPE=10` if 10 tape devices have been defined.
	■ %%DEV_TYPE_*ddd*
	Generic name of a device type, where *ddd* is the device address.
	For example:
	`%%DEV_TYPE_480=TAPE`
	`%%DEV_TYPE_490=CART`
	■ %%DEV_STAT_*ddd*
	Status of a device, where *ddd* is the device address.
	For example:
	`%%DEV_STAT_480=ONLINE`
	■ %%ID_*sfmid*
	Character used by CONTROL-M as a system identifier (set in AutoEdit System variable %%$SIGN) in a multi-system environment, where *smfid* is the SMF ID of the CPU.
	For example:
	`%%ID_SYS1=A`
	`%%ID_ESA1=B`

Rule Logic	This rule dynamically builds the set of Global variables that define the devices controlled by CONTROL-O and sets the status of these devices. These variables are used by other rules in this solution to update the Quantitative resource according to the device address. This rule initializes these Quantitative resources to the number of devices that are actually online. Changes made in this rule take effect when the rule is scheduled again. To remove previously defined device-type, Remove Global variables from CONTROL-O both from rule definitions and from the CONTROL-O $GLOBAL member.

Table 45 Initialize Device Global Variables Rule Structure (part 4 of 4)

Item	Description

Rule Actions	■ Initializes the device names, the associated resource names and their counters.
	■ Issues command D U,,,000,9999 in command-response mode, in order to get a list of all local devices and their statuses. Each line of the response (message IEE450I) contains two devices that are checked in a loop. The lines are checked for the type, status and count of all the devices whose generic names (%%DEVNAME_*type* variable) were defined in the rule.
	■ Sets Global variable %%DEV_INDEX_*index* to a device address.
	■ Sets Global variable %%DEV_TYPE_*ddd* to the generic name of the device type.
	■ Sets Global variable %%DEV_STAT_*ddd* to ONLINE or OFFLINE.
	For example:
	`%%DEV_INDEX_1=480`
	`%%DEV_TYPE_480=TAPE`
	`%%DEV_STAT_480=ONLINE`
	■ Sets Global variable %%DEV_TOTAL to the total number of devices of all types predefined in the rule and calculates the total number of devices of each type (%%DEV_T_*type*) and total number of devices of each type that are online (%%DEV_O_*type*).
	■ Sets the Quantitative resources for the defined device types to the calculated values.
	■ Issues informative messages showing the devices and values of the various counters.

Activating the Rule	Once ordered, the rule is executed.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—this rule is provided as an example. Implementation of the rule requires some customization.
Customization	■ Adapt the rule to site requirements by changing the Global Variable list to reflect the actual device definitions. ■ Check Global variables %%DEVNAME_*type*, %%RESNAME_*name* and %%DEV_*index*, which are set by the rule and adapt them to the site's definitions. The character used for %%ID_*smfid* in this rule is set in rule JES2GLBL in SolveWare Initialization table INITSLV. It may also require customization (see "SolveWare Initialization" on page 31).

Table 46 Device Brought Online Rule Structure (part 1 of 2)

Item	Description
Title	**Device Brought Online**
Name	**IEE302I**
Table	**DEVICES**
Message	`IEE302I {PATH(`*`ddd`*`,`*`xx`*`) | `*`ddd`*`} ONLINE` when the message does not contain the string PATH
Message Description	Device *ddd* was brought online.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Global Variables	■ %%DEV_TYPE_*ddd*
	Generic name of a device type, where *ddd* is the device address.
	For example:
	`%%DEV_TYPE_480=TAPE`
	`%%DEV_TYPE_490=CART`
	■ %%DEV_STAT_*ddd*
	Status of a device, where *ddd* is the address. Valid values: ONLINE and OFFLINE.
	For example:
	`%%DEV_STAT_480=ONLINE`
	■ %%DEV_O_*type*
	Total number of devices of the specified type that are online.
	For example:
	`%%DEV_O_CART=0` if there are no online cartridges
	`%%DEV_O_TAPE=5` if there are five tapes online
	■ %%ID_*sfmid*
	Character used by CONTROL-M as a system identifier (set in AutoEdit System variable %%$SIGN) in a multi-system environment, where smfid is the SMF ID of the CPU.
	For example:
	`%%ID_SYS1=A`
	`%%ID_ESA1=B`

Rule Logic	When a device is brought online in the system, the CONTROL-M Resources file requires adjustment to indicate the additional available resource (device). However, when the message indicates a PATH is being brought online, the message is ignored. The rule stores the current status (ONLINE or OFFLINE) of each device in Global variable %%DEV_STAT_*ddd*. If message IEE320I is issued for a device that is online, the message is ignored.

Table 46 Device Brought Online Rule Structure (part 2 of 2)

Item	Description

Rule Actions	■ Checks the device number in the Global Variable list to determine the name of the Quantitative resource to which the device refers.
	For example:
	`3490`
	`TAPE`
	■ Checks the status of the device (ONLINE or OFFLINE). If the message is issued for a device that is already online, the message is ignored.
	■ If the resource name is defined in the Global Variable list and %%DEV_STAT_*ddd* contains value OFFLINE, the rule increases the Quantitative resource by one in the CONTROL-M Resources file, sets Global variable %%DEV_STAT_*ddd* to ONLINE and increases Global variable %%DEV_O_*type* by one.

Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is 3—this rule is provided as an example. Implementation of the rule requires some customization.
Customization	The rule uses Global variables defined by the DEVTYPES rule (see “” on page 88), and by the JES2GLBL rule (see the discussion on SolveWare initialization rules on page 32). For the necessary customization, refer to those rules in this guide.

Table 47 Device Brought Offline Rule Structure (part 1 of 3)

Item	Description
Title	**Device Brought Offline**
Name	**IEF281I**
Table	**DEVICES**
Message	`IEF281I` *`ddd`* `NOW OFFLINE { - DEVICE IS BOXED}`
Message Description	Device *ddd* is now offline.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Table 47 Device Brought Offline Rule Structure (part 2 of 3)

Item	Description

Global Variables	■ %%DEV_TYPE_*ddd*
	Generic name of a device type, where *ddd* is the device address.
	For example:
	`%%DEV_TYPE_480=TAPE`
	`%%DEV_TYPE_490=CART`
	■ %%DEV_STAT_*ddd*
	Status of a device, where *ddd* is the device address. Valid values: ONLINE and OFFLINE.
	For example:
	`%%DEV_STAT_480=ONLINE`
	■ %%DEV_O_*type*
	Total number of devices of the specified type that are online.
	For example:
	`%%DEV_O_CART=0` if there are no online cartridges
	`%%DEV_O_TAPE=5` if there are five tapes online
	■ %%ID_*sfmid*
	Character used by CONTROL-M as a system identifier (set in AutoEdit System variable %%$SIGN) in a multi-system environment, where *smfid* is the SMF ID of the CPU.
	For example:
	`%%ID_SYS1=A`
	`%%ID_ESA1=B`

Rule Logic	When a device is brought offline in the system, the CONTROL-M Resources file requires adjustment to reflect that the resource (device) is no longer available. Global variable %%DEV_STAT_*ddd* is set to OFFLINE.

Rule Actions	■ Checks the device number in the Global Variable list to determine the name of the quantitative resource to which the device refers.
	For example:
	`3490`
	`TAPE`
	
	■ If the resource name is defined in the Global Variable list, the rule decreases the Quantitative resource by one in the CONTROL-M Resources file, sets Global variable %%DEV_STAT_*ddd* to OFFLINE and decreases %%DEV_O_*type* by one.

Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.

Table 47 Device Brought Offline Rule Structure (part 3 of 3)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—this rule is provided as an example. Implementation of the rule requires some customization.
Customization	The rule uses Global variables defined by the DEVTYPES rule (see “” on page 88), and by the JES2GLBL rule (see the discussion on SolveWare initialization in Chapter 2, “STARTSYS”). For the necessary customization, refer to those rules in this guide.

Table 48 Device Mount Request Rule Structure (part 1 of 3)

Item	Description
Title	**Device Mount Request**
Name	**IEC501?**
Table	**DEVICES**

Message	Any of the following messages:
	
	`IEC501A M ddd,ser{,labtype}{,den},jjj,sss,dsn`
	`IEC501E M ddd,ser{,labtype}{,den},jjj,sss,dsn`
	`IEF233A M ddd,ser{,labtype},jjj,sss,dsn`
	`IEF233D M ddd,ser{,labtype},jjj,sss,dsn`

Message Description	This message appears when a volume is to be mounted on a device.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Table 48 Device Mount Request Rule Structure (part 2 of 3)

Item	Description

Global Variables	■ %%DEV_TYPE_*ddd*
	Generic name of a device type, where *ddd* is the device address.
	For example:
	`%%DEV_TYPE_480=TAPE`
	`%%DEV_TYPE_490=CART`
	■ %%RESNAME_*name*
	Name of the Quantitative resource used by CONTROL-M for the device it controls.
	For example:
	`%%RESNAME_TAPE=TAPE %%RESNAME_CART=CARTRIDGE`
	■ %%DEV_STAT_*ddd*
	Status of a device, where *ddd* is the device address.
	For example:
	`%%DEV_STAT_480=ONLINE`
	■ %%ID_*sfmid*
	Character used by CONTROL-M as a system identifier (set in AutoEdit System variable %%$SIGN) in a multi-system environment, where *smfid* is the SMF ID of the CPU.
	For example:
	`%%ID_SYS1=A`
	`%%ID_ESA1=B`

Rule Logic	When a mount request for a device that is handled by CONTROL-O and is marked ONLINE, is displayed on the console, this rule decreases the device's Quantitative resource by one to reflect the occupied device and sets %%DEV_STAT_*ddd* to ONLINE_*jobid* (where *ddd* is the device address and *jobid* identifies the job that requested mounting of the device). If the device is marked OFFLINE, (for example, because the device was offline and was brought online implicitly by specifying the device name in the reply to message IEF238D), then a command to vary the device online is issued in order to trigger rule IEE302I. If the mount message that triggered the rule was issued by a job submitted by CONTROL-M, CONTROL-O automatically checks whether CONTROL-M has already allocated the resource to this job. If CONTROL-M has already allocated the resource to the job, the DO RESOURCE action is ignored. This check involves no special considerations in the rule definition.

Table 48 Device Mount Request Rule Structure (part 3 of 3)

Item	Description
Rule Actions	Checks the device number in the Global Variable list to determine the name of the Quantitative resource to which the device refers. For example: ■ 3490 ■ TAPE ■ If the resource name is defined in the Global Variable list and %%DEV_STAT_*ddd* is ONLINE, the rule does the following: ■ decreases the Quantitative resource in the CONTROL-M Resources file by one ■ decreases %%DEV_O_*ddd* by one ■ sets Global variable %%DEVALLOC_*jobid* to YES (to indicate that rule $HASP395 handles this job) ■ sets %%DEV_STAT_*ddd* to ONLINE_*jobid* (to specify that the device is used by that job) ■ If the device status is OFFLINE, command V *ddd*,ONLINE is issued to trigger rule IEE302I that varies online devices.
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—this rule is provided as an example. Implementation of the rule requires some customization.
Customization	The rule uses Global variables defined by the DEVTYPES rule (see “” on page 88), and by the JES2GLBL rule (see the discussion on SolveWare initialization in Chapter 2, “STARTSYS”). For the necessary customization, refer to those rules in this guide.

Table 49 Device Demount Request Rule Structure (part 1 of 3)

Item	Description
Title	**Device Demount Request**
Name	**IEC234E**
Table	**DEVICES**

Message	Either of the following messages:
	
	`IEF234E {K | D | R} ddd, ser,{PVT | PUB | STR},{jjj,sss}{,SPACE=prm}`
	
	`IEC502E n ddd,ser{,labtype}{ddn-c} {SPACE=ccc,ttt,aaa/ccc,ttt}, jjj,sss{,dsn}`

Message Description	This message appears when a volume is to be demounted from a device.

Table 49 Device Demount Request Rule Structure (part 2 of 3)

Item	Description
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Global Variables	■ %%DEV_TYPE_*ddd*
	Generic name of a device type, where *ddd* is the device address.
	For example:
	`%%DEV_TYPE_480=TAPE`
	`%%DEV_TYPE_490=CART`
	■ %%RESNAME_*name*
	Name of the Quantitative resource used by CONTROL-M for the device it controls.
	For example:
	`%%RESNAME_TAPE=TAPE %%RESNAME_CART=CARTRIDGE`
	■ %%DEV_O_*type*
	Total number of devices of the specified type that are online.
	For example:
	`%%DEV_O_CART=0` if there are no online cartridges
	`%%DEV_O_TAPE=5` if there are five tapes online
	■ %%DEV_STAT_*ddd*
	Status of a device, where *ddd* is the device address. Valid values: ONLINE and OFFLINE.
	For example:
	`%%DEV_STAT_480=ONLINE`
	■ %%ID_*sfmid*
	Character used by CONTROL-M as a system identifier (set in AutoEdit System variable %%$SIGN) in a multi-system environment, where *smfid* is the SMF ID of the CPU.
	For example:
	`%%ID_SYS1=A`
	`%%ID_ESA1=B`

Rule Logic	When a demount request appears on the console, this rule increases the device's Quantitative resource by one to reflect the occupied device and sets %%DEV_STAT_*ddd* to OFFLINE (where *ddd* is the device address). If the demount message that triggered the rule was issued by a job submitted by CONTROL-M, CONTROL-O automatically checks whether CONTROL-M has already allocated the resource to this job. If CONTROL-M has already allocated the resource to the job, the DO RESOURCE action is ignored. This check involves no special considerations in the rule definition.

Table 49 Device Demount Request Rule Structure (part 3 of 3)

Item	Description
Rule Actions	Checks the device number in the Global Variable list to determine the name of the Quantitative resource to which the device refers. For example: ■ 3490 ■ TAPE ■ If the resource name is defined in the Global Variable list, the rule ■ increases the Quantitative resource in the CONTROL-M Resources file by one ■ sets %%DEV_STAT_*ddd* to ONLINE
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—this rule is provided as an example. Implementation of the rule requires some customization.
Customization	The rule uses Global variables defined by the DEVTYPES rule (see “” on page 88), and by the JES2GLBL rule (see the discussion on SolveWare initialization in Chapter 2, “STARTSYS”). For the necessary customization, refer to those rules in this guide.

Table 50 Job Ended Release Device Quantitative Resources Rule Structure (part 1 of 3)

Item	Description
Title	**Job Ended Release Device Quantitative Resources**
Name	**$HASP395**
Table	**STARTSYS**
Message	`$HASP395 jjj ENDED`
Message Description	Job *jjj* finished executing.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Table 50 Job Ended Release Device Quantitative Resources Rule Structure (part 2 of 3)

Item	Description

Global Variables	■ %%DEVALLOC_*jobid*
	Value is YES when at least one device is held by the job identified by *jobid*.
	■ %%DEV_*index*
	Generic device name, indexed by a sequence number.
	For example:
	`%%DEV_1=TAPE`
	`%%DEV_2=CART`
	■ %%DEV_TOTAL
	Total number of all devices units types controlled by CONTROL-O.
	For example:
	`%%DEV_TOTAL=20`
	■ %%DEV_INDEX_*index*
	Device unit address, indexed by a sequence number.
	For example:
	`%%DEV_INDEX_1=460`
	`%%DEV_INDEX_2=461`
	■ %%DEV_TYPE_*ddd*
	Generic name of a device type, where *ddd* is the device address.
	For example:
	`%%DEV_TYPE_480=TAPE`
	`%%DEV_TYPE_490=CART`
	■ %%DEV_STAT_*ddd*
	Status of a device, where *ddd* is the device address. Valid values for this variable are ONLINE and OFFLINE.
	For example:
	`%%DEV_STAT_480=ONLINE`
	■ %%ID_*sfmid*
	Character used by CONTROL-M as a system identifier (set in AutoEdit System variable %%$SIGN) in a multi-system environment, where *smfid* is the SMF ID of the CPU.
	For example:
	`%%ID_SYS1=A`
	`%%ID_ESA1=B`

Table 50 Job Ended Release Device Quantitative Resources Rule Structure (part 3 of 3)

Item	Description
Rule Logic	This rule handles situations where a mount message issued for a job is not followed by an appropriate demount message (for example, when a scratch tape is mounted but not opened, no demount message is issued). In this case, the device Quantitative resource is not released, since the demount message did not appear. For each job running in the system, the MOUNT rule sets the corresponding Global variable %%DEVALLOC_*jobid* to YES if at least one device was mounted by the job. In this case, the status of the allocated devices (%%DEVSTAT_ddd) becomes ONLINE_*jobid*. The rule checks the %%DEVALLOC_*jobid* variable to determine whether the rule processes the job (identified by *jobid*). If its value is YES, all the devices with status ONLINE_*jobid* is counted by device name and their statuses are reset to ONLINE. The Quantitative resource device counters are increased according to their respective device names. The Global %%DEVALLOC_*jobid* is set to %%UNDEF, which removes it from the Global Variable list. When all devices have been released from the job by the DEMOUNT rule, no devices are found.
Rule Actions	If there are devices allocated by the job, the rule performs the following: ■ counts the allocated devices that were not handled by the DEMOUNT rule ■ resets Global variable %%DEVSTAT_*ddd* to ONLINE ■ deletes Global variable DEVALLOC_*jobid* ■ increases the device's Quantitative resource in accordance with the value of the counters
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of this rule requires some customization.
Customization	The rule is defined to handle Quantitative resources TAPE and CASSET. Update these values. You can add additional Quantitative resources to conform to site requirements.

Table 51 Device Swap in Progress Rule Structure

Item	Description
Title	**Device Swap in Progress**
Name	**IGF502E**
Table	**DEVICES**
Message	`IGF502E PROCEED WITH SWAP OF ddd1 TO ddd2`
Message Description	A device swap is in progress. The tape or cartridge volume is moved from device ddd1 to device ddd2.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Global Variables	■ %%DEV_TYPE_*ddd*
	Generic name of a device type, where *ddd* is the device address.
	For example:
	`%%DEV_TYPE_480=TAPE`
	`%%DEV_TYPE_490=CART`
	
	■ %%DEV_STAT_*ddd*
	Status of a device, where *ddd* is the device address.
	For example:
	`%%DEV_STAT_480=ONLINE`

Rule Logic	If the device types of devices *ddd1* and *ddd2* are defined to the SolveWare, target device *ddd2* is assigned the current status of source device *ddd1* and the status of source device *ddd1* is reset to ONLINE.
Rule Actions	The value of Global variable %%DEV_STAT_*ddd2* is set to the value of %%DEV_STAT_*ddd1*. The value of Global variable %%DEV_STAT_*ddd* is set to ONLINE.
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1—little or no customization is required before implementation.

Device-Not-Available Handling—Option 1

This solution handles situations in which an attempt to allocate a device fails. A message sent to the console asks for the operator's instructions – whether to allocate a different device, cancel the job, or wait until the device becomes available. The WAIT option appears only if MVS determines that it is reasonable to wait for the device.

When a message's text indicates that the WAIT option is valid, this solution instructs MVS to WAIT. When MVS issues a message asking the operator whether to hold the units already allocated to the waiting job, this solution instructs MVS not to hold the units (that is, replies NOHOLD).

Rules

The Device-Not-Available Handling—Option 1 solution includes the Device-Not-Available Handling—Option 1 rule.

Rules Structure

The following tables describe the structure of the Device-Not-Available Handling—Option 1 solution rules.

Table 52 Device-Not-Available—Option 1 Rule Structure (part 1 of 2)

Item	Description
Title	**Device-Not-Available—Option 1**
Name	**IEF238D**
Table	**DEVICES**
Message	`IEF238D jjj - REPLY {DEVICE NAME} {,}{'WAIT'} OR 'CANCEL'` when the message contains the string WAIT.
Message Description	Device allocation cannot be performed (completed) by MVS for specified jobname *jjj*.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.

Table 52 Device-Not-Available—Option 1 Rule Structure (part 2 of 2)

Item	Description
Rule Logic	If a device is currently not available and the WAIT string appears in the message text, the rule replies WAIT, which instructs MVS to have the job wait for the devices to become available. The rule then expects MVS to issue message IEF433D, which asks whether to hold already allocated devices. Since holding already allocated devices can result in a deadlock, the rule replies NOHOLD.
Rule Actions	■ Sets AutoEdit reserved user-defined variable %%RESPMSG to IEF433D. ■ Replies WAIT to the message, using the message reply number. ■ If a response message (IEF433D) is intercepted, replies to that message with NOHOLD.
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1— little or no customization is required before implementation.

Device-Not-Available Handling—Option 2

This solution handles situations in which an attempt to allocate a device fails. A message sent to the console asks for the operator's instructions – whether to allocate a different device, cancel the job, or wait until the device becomes available. The WAIT option appears only if MVS determines that it is reasonable to wait for the device.

When the MVS message text indicates that the WAIT option is valid, this solution instructs MVS to WAIT. When MVS then issues a message asking the operator whether to hold, the units already allocated to the waiting job, this solution instructs MVS not to hold the units (that is, replies NOHOLD).

When the MVS message text does not contain the WAIT option, a distinction is made between test and production jobs. If the job that failed to allocate a device is a test job, the rule replies CANCEL. If the job is a production job, no reply is issued, but a message is sent to the operator explaining the situation.

Rules

The Device-Not-Available Handling—Option 2 solution includes the Device-Not-Available Handling—Option 2 rule.

Rules Structure

The following tables describe the structures of the Device-Not-Available Handling—Option 2 solution rules.

Table 53 Device-Not-Available—Option 2 Rule Structure (part 1 of 2)

Item	Description
Title	**Device Not-Available Handling—Option 2**
Name	**IEF238D**
Table	**DEVICES2**
Message	`IEF238D jjj - REPLY {DEVICE NAME} {,}{'WAIT'} OR 'CANCEL'`
Message Description	Device allocation cannot be performed (completed) by MVS for specified jobname *jjj*.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	If the message appeared with the WAIT option, the reply is answered WAIT. The rule then expects MVS to issue message IEF433D, which asks whether to hold already allocated devices. Since holding already allocated devices can result in a deadlock, the rule replies NOHOLD. If the WAIT option does not appear and the job is a test job, the rule replies CANCEL. The owner of the test job must then investigate the situation. If the WAIT option does not appear and the job is a production job, the rule sends a highlighted message to the operator console prompting the operator to take action. The production job awaits the operator's actions and is not automatically canceled.
Rule Actions	If the message contains the WAIT option, the rule performs the following actions: ■ sets AutoEdit reserved user-defined variable %%RESPMSG to IEF433D ■ replies WAIT to the message, using the message reply number ■ if a response message (IEF433D) is intercepted, replies to that message with NOHOLD If the message does not contain the WAIT option, the rule performs the following actions: ■ for a test job, replies CANCEL to the message using the message reply number ■ for a production job, sends a message to the operator

Table 53 Device-Not-Available—Option 2 Rule Structure (part 2 of 2)

Item	Description
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—this rule is provided as an example. Implementation of the rule requires some customization.
Customization	This rule identifies test and production jobs by prefix. All test jobs are assumed to have a prefix of TEST; all jobs with a prefix other than TEST are assumed to be production jobs. Adapt these values to reflect site job prefix conventions.

Device-Not-Available Handling in a Multisystem Environment—Option 3

This solution enables tape or cartridge devices to be shared among multiple MVS systems, without requiring operator intervention, by automatically switching devices between systems when necessary. This solution is designed for JES2 sites.

When allocation of a device fails, MVS message IEF290E is issued identifying other eligible devices that are offline and can potentially be allocated for the job.

A second message, IEF238D is sent to the console requesting (from the operator) instructions on how to proceed. One of the following replies can be specified:

- device name (address of a different device for allocation)
- WAIT (until a device becomes available. This option appears only if MVS determines that it is reasonable to wait for a device)
- CANCEL (the job)

Normally an operator checks whether a device of the same type is available (that is, online and not in use) on another system.

- If it is, the operator varies the device offline from the other system and replies to message IEF238D with the device address (for allocation).
- If it is not, the operator usually responds WAIT (if this option is available) or CANCEL (if the WAIT option is not available).

This solution performs the same actions automatically, as follows:

- The first rule in this solution issues a request for a device to the other systems one by one, until a device is found or all systems have been queried.
- The second rule in this solution is activated in each system that is queried. This rule determines if a device is available in the system, and if so, the rule varies the device offline and provides the device address to the first rule.

The first rule replies to the message based on the results of the query.

Rules

The Device-Not-Available Handling—Option 3 solution includes:

- Device-Not-Available in a Multisystem Environment—Option 3
- Check for Free Online Device
- Suppress CONTROL-O Internal Command

Rules Structure

The following tables describe the structures of the Device-Not-Available Handling—Option 3 solution rules.

Table 54 Device-Not-Available in a Multisystem Environment—Option 3 Rule Structure (part 1 of 4)

Item	Description
Title	**Device-Not-Available in a Multisystem Environment—Option 3**
Name	**IEF290E**
Table	**DEVICES3**

Table 54 Device-Not-Available in a Multisystem Environment—Option 3 Rule Structure (part 2 of 4)

Item	Description

Message	`IEF290E/IEF287E/IEF877E`
	`jjj stepname dsname` `NEEDS # UNITS FOR` *`volume`*
	`########`
	`OFFLINE not accessible`
	`devname1=ddd1,ddd2,...`
	`devname2=ddd3,ddd4,..`
	`devname3=.......`
	
	where:
	
	jjj is the jobid
	devname1, *devname2*,... are device names
	ddd1, *ddd2*, ... are addresses of OFFLINE devices
	dd3,*dd4*,... are addresses of non-accessible devices

Message Description	Device allocation cannot be performed (completed) by MVS for specified job *jjj*. The following (if any) devices are OFFLINE or cannot be accessed by MVS.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Global Variables	■ %%DEV-TYPE_*ddd*
	Generic name of a device type, where *ddd* is the device address.
	For example:
	`%%DEV_TYPE_480=TAPE`
	`%%DEV_TYPE_490=CART`
	■ %%DEV_STAT_*ddd*
	Status of a device, where *ddd* is the device address. Valid values: ONLINE and OFFLINE.
	For example:
	`%%DEV_STAT_480=ONLINE`
	■ %%DEV_*index*
	Generic device name, indexed by a sequence number.
	For example:
	`%%DEV_1=TAPE`
	`%%DEV_2=CART`
	■ %%DEV_INDEX_*index*
	Device unit address, indexed by a sequence number.
	For example:
	`%%DEV_INDEX_1=460`
	`%%DEV_INDEX_2=461`
	■ %%JES2_ROUTE_*smf*
	Multi-access spool member, multi-access spool member node, or node member used to send a command over the network to the specified system. *smf* is the SMF ID of the CPU.
	For example:
	`%%JES2_ROUTE_ESA1=$M2 %%JES2_ROUTE_ESA2=$N2M3`

Table 54 Device-Not-Available in a Multisystem Environment—Option 3 Rule Structure (part 3 of 4)

Item	Description
Rule Logic	If message IEF290E appears with a list of OFFLINE devices, this rule checks whether the requested device is a tape or cartridge. If the requested device is a tape or cartridge, this rule waits for the issuance of message IEF238D, which requests instructions on how to proceed. Command rule CTOREQ (see the following section) is then activated on the other MVS system to look for an available device of the same type. (If rule CTOREQ finds an available device, it varies the device offline on the other MVS system and passes the device address to this rule.) If this rule receives a device address from rule CTOREQ, it replies to message IEF238D with the device address and CONTROL-O varies the device online for this system. If this rule does not receive a device address from rule CTOREQ, it issues a WAIT reply if this was one of the message options. If WAIT was not a message option, this rule replies CANCEL to message IEF238D.
Rule Actions	If message IEF290E contains a list of offline units, this rule determines if the requested device is a tape or cartridge. If the device is a tape or cartridge, the rule ■ waits for message IEF238E and stores the reply %%REPLY in AutoEdit variable %%R. ■ waits until a device address is received from Command rule CTOREQ, or until all MVS systems with a CONTROL-O monitor have been checked for an available device This rule issues the following command, system by system, in command-response mode. The command is passed to the appropriate system using NJE. `CTOREQ GETUNIT %%SMFID %%CONTROLO %%JOBID` *`unittype`* `1` If a device address is received from rule CTOREQ using NJE message $HASP249, this rule answers to message IEF238D with the device address. If no device address is received from CTOREQ, or if there were no tape or cartridge devices listed by message IEF290E, then if WAIT is a valid reply option to message IEF238D, this rule replies WAIT. Otherwise, this rule replies CANCEL.
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.

Table 54 Device-Not-Available in a Multisystem Environment—Option 3 Rule Structure (part 4 of 4)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	The rule uses Global variables defined by the DEVTYPES rule (see “” on page 88), and by the JES2GLBL rule (see the discussion on SolveWare initialization in Chapter 2, “STARTSYS”). For the necessary customization, refer to those rules in this guide.

Table 55 Check for Free Online Device Rule Structure (part 1 of 3)

Item	Description
Title	**Check for Free Online Device**
Name	**CTOREQ**
Table	**DEVICES3**
Command	`CTOREQ GETUNIT` *`smfid monitor-name jobid unit-type`* `1`
Command Description	This command is issued by rule IEF290E in this solution to query for an available device of the specified type.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Table 55 Check for Free Online Device Rule Structure (part 2 of 3)

Item	Description

Global Variables	■ %%DEV_TYPE_*ddd*
	Generic name of a device type, where *ddd* is the device address.
	For example:
	`%%DEV_TYPE_480=TAPE`
	`%%DEV_TYPE_490=CART`
	■ %%DEV_STAT_*ddd*
	Status of a device, where *ddd* is the device address. Valid values: ONLINE and OFFLINE.
	For example:
	`%%DEV_STAT_480=ONLINE`
	■ %%DEV_*index*
	Generic device name, indexed by a sequence number.
	For example:
	`%%DEV_1=TAPE`
	`%%DEV_2=CART`
	■ %%JES2_ROUTE_*smf*
	Multi-access spool member, multi-access spool member node, or node member used to send a command over the network to the specified system. smf is the SMF ID of the CPU.
	For example:
	`%%JES2_ROUTE_ESA1=$M2 %%JES2_ROUTE_ESA2=$N2M3`
	■ %%DEV_TOTAL
	The total number of device units of all types controlled by CONTROL-O.
	For example:
	`%%DEV_TOTAL=20`

Rule Logic	This rule receives a CTOREQ command using NJE from rule IEF290E. The CTOREQ command requests an available (that is, free, online) device of the specified type from the MVS system in which this rule is operating. This rule then scans the Global variables %%DEV_TYPE_*ddd* looking for devices of the type requested. If such a device is found, the rule checks the corresponding Global variable %%DEV_STAT_*ddd* to determine if the device is online and free. If the device is online and free, the rule varies the device offline and passes the device address to rule IEF290E.
Rule Actions	Checks for an online device of the requested type. If an online device of the correct type is found, the rule does the following: ■ issues the command `V ddd,OFFLINE` to vary the device offline, and waits for completion. ■ sends the device address to the requesting system by issuing the following command: `%%JES2_ROUTE_%%SMF;%%RPLY`

Table 55 Check for Free Online Device Rule Structure (part 3 of 3)

Item	Description
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	The rule uses Global variables defined by the DEVTYPES rule (see “” on page 88), and by the JES2GLBL rule (see the discussion on SolveWare initialization in Chapter 2, “STARTSYS”). For the necessary customization, refer to those rules in this guide.

Table 56 Suppress CONTROL-O Internal Command Rule Structure

Item	Description
Title	Suppress CONTROL-O Internal Command
Name	CTO*
Table	DEVICES3
Command	CTO* (all commands that start with CTO)
Command Description	Internal commands starting with CTO are used in this solution to communicate between rules on different MVS systems.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	Rules CTOREQ and IEE290E in this solution communicate with each other by issuing internal commands prefixed CTO. These commands normally cause issuance of message IEE305I, which indicates that the commands are invalid (that is, not recognized by MVS). This rule suppresses the commands so that the message IEE305I is not issued.
Rule Actions	Suppresses commands starting with CTO.
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1—little or no customization is required before implementation.

LOGREC

LOGREC solutions are designed to automate management of the Environmental Record Editing and Printing (EREP) facility. EREP is a diagnostic facility that reads and edits environmental records generated by both software and hardware.

Information stored in EREP records can be crucial when trying to locate problems. Therefore, handle EREP errors as soon as they occur to prevent the loss of environmental information.

NOTE

All messages (DO SHOUT actions) in the LOGREC rules are sent to an INCONTROL user named U-SYSADMIN. Define a user with this name in the IOA Dynamic Destination table (CTMDEST). For more information, see the Dynamic Destination Table chapter of the *INCONTROL for z/OS Administrator Guide*.

In many cases, rule definitions make use of the inverse IN condition feature. This feature activates rules only if the specified IN conditions are not set. For more information about using inverse IN conditions, see the section on SolveWare implementation considerations in Chapter 1, “Introduction.”

Solutions Provided

SolveWare subject LOGREC contains the following solutions:

- Copy and Clean SYS1.LOGREC

  Handles situations where the EREP dataset SYS1.LOGREC fills up. A copy or clean job is automatically sent to prevent loss of environmental information.

- EREP Problem Alerts

Intercepts error messages issued by EREP and notifies the system administrator of the problems.

- LOGREC Rule Thresholds

  Handles exceeded thresholds of all other LOGREC rules.

Copy and Clean SYS1.LOGREC

EREP stores environmental records in dataset SYS1.LOGREC. When this dataset is nearly full, a message appears on the system console. If SYS1.LOGREC is not cleaned, another message is issued when SYS1.LOGREC becomes full.

To prevent loss of environmental records, the following solution is triggered by both the SYS1.LOGREC NEAR FULL and SYS1.LOGREC IS FULL messages. The 'FULL' message is defined for any situations in which the 'NEAR FULL' message was disregarded.

Utility program IFCEREP1 is initiated to copy and clean SYS1.LOGREC when it becomes nearly full. EREP records can be copied to a generation dataset, to a single cumulative dataset (with DISP=MOD), and so on.

Rules

The Copy and Clean SYS1.LOGREC solution includes the following rules:

- **SYS1.LOGREC IS NEAR FULL**
- **SYS1.LOGREC IS FULL**

Rules Structure

The following tables describe the structure of the Copy and Clean SYS1.LOGREC solution rules.

Table 57 SYS1.LOGREC Is Near Full, SYS1.LOGREC Is Full Rule Structure (part 1 of 3)

Item	Description
Title	**SYS1.LOGREC is Near Full, SYS1.LOGREC is Full**
Name	**IFB060E**
Table	**LOGREC**

Table 57 SYS1.LOGREC Is Near Full, SYS1.LOGREC Is Full Rule Structure (part 2 of 3)

Item	Description

Message	One of the following messages:
	
	`IFB060E SYS1.LOGREC NEAR FULL`
	`IFB040I SYS1.LOGREC AREA IS FULL, hh:mm:ss`
	`IFB080E LOGREC DATA SET NEAR FULL, DSN=dsn`
	`IFB081I LOGREC DATA SET IS FULL,hh.mm.ss, DSN=dsn`

Message Description	Dataset SYS1.LOGREC has reached 90% of its capacity, or the SYS1.LOGREC area is full and cannot hold more information.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	The rule is triggered when dataset SYS1.LOGREC is either full or nearly full. Either situation normally requires a user response to initiate a job or started task that executes program IFCEREP1 to dump and clean SYS1.LOGREC. The rule adds condition or date CTO-LOGRECCP-GO STAT, which triggers a pre-scheduled cyclic job in CONTROL-M to dump and clean SYS1.LOGREC. (For more details on the job scheduling definition, see Customization in this table.) Once the rule has been triggered, it is temporarily deactivated by adding its inverse IN condition or date CTO-IFB060E-HANDLED 0101. This prevents multiple triggering of the rule caused by the additional appearances of FULL or NEAR FULL messages before the job has finished cleaning SYS1.LOGREC. Once the job has successfully executed, condition CTO-IFB060E-HANDLED is deleted to reactivate the rule. Condition CTO-LOGRECCP-GO, which triggered the pre-scheduled job, is deleted upon completion of the job.
Rule Actions	■ Determines the text of the message to be sent to the system administrator according to the type of message. ■ Sends a message notifying user U-SYSADMIN of both the occurrence of the problem and the corrective steps that were performed. ■ Sets condition CTO-IFB060E-HANDLED to trigger a pre-scheduled job in CONTROL-M that copies and cleans SYS1.LOGREC. ■ Sets condition CTO-LOGRECCP-GO to deactivate the rule temporarily (see Rule Logic in this table).

Table 57 SYS1.LOGREC Is Near Full, SYS1.LOGREC Is Full Rule Structure (part 3 of 3)

Item	Description
Activating the Rule	Once ordered, the rule remains active until one of the messages IFB060E or IFB040I or IFB080E or IFB081I exceeds a predefined threshold. (For more information regarding threshold handling, see "LOGREC Rule Thresholds" on page 119.) The rule is also temporarily deactivated when it is triggered and reactivated after the copy or clean job finishes OK (see Rule Logic in this table).
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	A job scheduling definition and JCL for the SYS1.LOGREC copy or clean job must be created. The SOLVSCHD and SOLVJCL libraries contain a sample scheduling definition and JCL to copy or clean SYS1.LOGREC. These samples can be adapted to site conventions and requirements. Define the job (in the CONTROL-M job scheduling definition) as a cyclic job or cyclic started task with a MAXWAIT value of 99. Order the job once, but do not manually remove it from the CONTROL-M Active Jobs file. The MAXWAIT value of 99 ensures that the job is never removed from the Active Jobs file by the CONTROL-M New Day procedure. The cyclic job is always ready for submission. It is triggered by adding the prerequisite condition or date CTO-LOGRECCP-GO STAT. When an execution of the job is completed, this condition is deleted. This prevents cyclic re-invoking of the job and ensures that the job is only invoked again if the rule is triggered again.

EREP Problem Alerts

This solution handles various EREP error situations that are not resolved automatically. This solution ensures that the system administrator is notified immediately by CONTROL-O when EREP problems occur, so that the system administrator can locate the source of the error and solve the problem.

Rules

The EREP Problem Alerts solution includes the following rules:

- SYS1.LOGREC Access Failed
- SYS1.LOGREC Format Error
- SYS1.LOGREC Cannot Be Accessed

Rules Structure

The following tables describe the structure of the EREP Problem Alerts solution rules.

Table 58 STS1.LOGREC Access Failed Rule Structure

Item	Description
Title	**SYS1.LOGREC Access Failed**
Name	**IFB0301**
Table	**LOGREC**
Message	Either of the following messages: `IFB030I SYS1.LOGREC I/O ACCESS ERROR,`*`sens,stat,hh.mm.ss`* `IFB082I LOGREC DATA SET I/O ACCESS ERROR,`*`sens,stat,hh.mm.ss`*`, DSN=`*`dsn`*
Message Description	Dataset SYS1.LOGREC was accessed either to read or write a record and a non-correctable input/output error occurred.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	The rule is triggered when an I/O error occurs while trying to access dataset SYS1.LOGREC. CONTROL-O immediately brings the error to the attention of the system administrator who can then determine the cause of the problem and proceed to solve it.
Rule Actions	Sends a message notifying user U-SYSADMIN of the problem.
Activating the Rule	Once ordered, the rule remains active until message IFB030I or IFB082I exceeds a predefined threshold. (For more information regarding threshold handling, see "LOGREC Rule Thresholds" on page 119.)
Recommended Mode or Category	Activate this rule in PROD mode. The SolveWare category for this rule is 1—little or no customization is required before implementation.

Table 59 SYS1.LOGREC Format Error Rule Structure

Item	Description
Title	**SYS1.LOGREC Format Error**
Name	**IFB0501**
Table	**LOGREC**
Message	`IFB050I/IFB083I FORMAT ERROR,`*`hh.mm.ss`*
Message Description	The header record of dataset SYS1.LOGREC is missing or invalid.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	The rule is triggered when an error occurs while trying to access the SYS1.LOGREC header record. CONTROL-O immediately brings the error to the attention of the system administrator who can then determine the cause of the problem and proceed to solve it.
Rule Actions	Sends a message notifying user U-SYSADMIN of the problem.
Activating the Rule	Once ordered, the rule remains active until message IFB050I or IFB083I exceeds a predefined threshold. (For more information regarding threshold handling, see "LOGREC Rule Thresholds" on page 119.)
Recommended Mode or Category	Activate this rule in PROD mode. The SolveWare category for this rule is 1—little or no customization is required before implementation.

Table 60 SYS1.LOGREC Cannot Be Accessed Rule Structure (part 1 of 2)

Item	Description
Title	**SYS1.LOGREC Cannot Be Accessed**
Name	**IFB070I**
Table	**LOGREC**
Message	Either of the following messages: `IFB070I SYSRES CANNOT BE ACCESSED. RECORD IS LOST` `IFB084I LOGREC DATA SET CANNOT BE ACCESSED, RECORD IS LOST, DSN=`*`dsn`*
Message Description	The SYSRES (System Residence) volume is unavailable; therefore, SYS1.LOGREC cannot be accessed.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.

Table 60 SYS1.LOGREC Cannot Be Accessed Rule Structure (part 2 of 2)

Item	Description
Rule Logic	The rule is triggered when the SYSRES volume is unavailable for any reason. CONTROL-O immediately brings the error to the attention of the system administrator who can then determine the cause of the problem and proceed to solve it.
Rule Actions	Sends a message notifying user U-SYSADMIN of the problem.
Activating the Rule	Once ordered, the rule remains active until message IFB070I or IFB084I exceeds a predefined threshold. (For more information regarding threshold handling, see "LOGREC Rule Thresholds" on page 119.)
Recommended Mode or Category	Activate this rule in PROD mode. The SolveWare category for this rule is 1—little or no customization is required before implementation.

LOGREC Rule Thresholds

This solution handles message overload – situations in which a message appears on the console more times than is acceptable. If an EREP message appears too often on the system console, threshold rules deactivate the relevant LOGREC rule until the source of the problem is found and the problem corrected.

For more information regarding threshold handling, see "SolveWare Implementation Considerations" on page 24.

Rules

The LOGREC Rule Thresholds solution includes the following rules:

- Handling Exceeded LOGREC Thresholds
- Resetting LOGREC Rule Threshold Conditions

Rules Structure

The following tables describe the structure of the LOGREC Rule Thresholds solution rules.

Table 61 Handling Exceeded LOGREC Thresholds Rule Structure (part 1 of 2)

Item	Description
Title	**Handling Exceeded LOGREC Thresholds**
Name	**IFB0301**
Table	**LOGREC**
Message	Any of the following messages: IFB030I SYS1.LOGREC I/O ACCESS ERROR,*sens,stat,hh.mm.ss* IFB040I SYS1.LOGREC AREA IS FULL,*hh.mm.ss* IFB050I FORMAT ERROR,*hh.mm.ss* IFB060E SYS1.LOGREC NEAR FULL IFB070I SYSRES CANNOT BE ACCESSED. RECORD IS LOST IFB080E LOGREC DATA SET NEAR FULL, DSN=*dsn* IFB081I LOGREC DATA SET IS FULL,*hh.mm.ss*, DSN=*dsn* IFB082I LOGREC DATA SET I/O ACCESS ERROR,*sens,stat,hh.mm.ss*, DSN=*dsn* IFB083I LOGREC DATA SET FORMAT ERROR,*hh.mm.ss*, DSN=*dsn* IFB084I LOGREC DATA SET CANNOT BE ACCESSED, RECORD IS
Message Description	All messages handled by rules comprising the LOGREC solutions.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.

Table 61 Handling Exceeded LOGREC Thresholds Rule Structure (part 2 of 2)

Item	Description
Rule Logic	To avoid message overload situations, this rule deactivates LOGREC rules whose messages have exceeded a predetermined number of appearances in a period of time. These threshold values are defined for every message included in this rule in the threshold parameters APPEARED ### TIMES IN #### MINUTES. To synchronize threshold handling correctly, this rule is assigned a higher PRIORITY value than the message rules that it monitors, and has a CONTINUE SEARCH value of Y (Yes). Deactivation of a LOGREC rule is achieved by adding the appropriate (inverse) IN prerequisite condition that is defined for the rule. To reactivate a deactivated rule, the threshold conditions must be deleted. This can be done either manually or automatically by CONTROL-O (see the following section). Threshold conditions are specified in the IGNORE list of the CONTROL-M CONTDAY procedure (see "SolveWare Implementation Considerations" on page 24).
Rule Actions	■ Notifies user U-SYSADMIN that the message that exceeded its threshold is no longer handled by CONTROL-O. ■ Sets the appropriate IN condition to deactivate the rule. The format of this condition is: `CTO-msgid-THRESH` where *msgid* is the message ID of the specific message.
Activating the Rule	Once scheduled, the rule is triggered whenever one of the above messages exceeds its threshold.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Review each LOGREC message monitored by this rule and determine appropriate threshold values. For each message included in this rule, adapt to site requirements APPEARED ### TIMES IN #### MINUTES values, which specify a number of appearances in a time period.

Table 62 Resetting LOGREC Rule Threshold Conditions Rule Structure

Item	Description
Title	**Resetting LOGREC Rule Threshold Conditions**
Name	**RESLOGRC**
Table	**LOGREC**
Event	RESLOGRC
Event Description	This Event rule deletes all threshold conditions for all LOGREC rules.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	Using the INTERVAL parameter, this rule periodically deletes the threshold conditions of all LOGREC rules, to reactivate LOGREC rules that exceeded their thresholds. For more information about the resetting of threshold conditions, see Customization in this table.
Rule Actions	■ Deletes condition or date CTO-IFB030I-THRESH STAT ■ Deletes condition or date CTO-IFB040I-THRESH STAT ■ Deletes condition or date CTO-IFB050I-THRESH STAT ■ Deletes condition or date CTO-IFB060E-THRESH STAT ■ Deletes condition or date CTO-IFB070I-THRESH STAT
Activating the Rule	Once scheduled, the rule is triggered periodically according to the INTERVAL parameter specification.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Adapt the INTERVAL parameter to site requirements: If you use the INTERVAL parameter without the TIME FROM parameter, the threshold conditions are deleted when the rule is ordered.
	Note: Threshold conditions are also deleted at time of IPL. For more information, see the section on initializing SolveWare in Chapter 2, “STARTSYS.” Threshold conditions are specified in the IGNORE list of the CONTROL-M CONTDAY procedure. For details, see the section on initializing SolveWare in Chapter 2, “STARTSYS.”

SMF

Solutions in SolveWare subject SMF are designed to automate management of the System Management Facilities (SMF). SMF is a standard MVS feature that collects and records system and job related information. This information is kept in SMF datasets to be used later as input for user-written report programs.

NOTE

All messages (DO SHOUT actions) in SMF rules are sent to an INCONTROL user named U-SYSADMIN. A user with this name must be defined in the IOA Dynamic Destination table (CTMDEST). For more information, see the Dynamic Destination Table chapter of the *INCONTROL for z/OS Administrator Guide*.

In many cases, rule definitions make use of the inverse IN condition feature. This feature activates rules only if the specified IN conditions are not set. For more information about using inverse IN conditions, see the section on SolveWare implementation considerations in Chapter 1, “Introduction.”

Solutions Provided

SolveWare subject SMF contains the following solutions:

- Copy and Clean SYS1.MANx

  For situations where the SMF dataset SYS1.MANx becomes full. A copy and clean job is sent automatically to prevent loss of system information.

- SMF Problem Alerts

  Manages error messages issued by SMF. A message is sent notifying the system administrator of the problem.

- SMF Rule Thresholds

  Handles exceeded thresholds of all other SMF rules.

Copy and Clean SYS1.MAN*x*

SMF datasets are named SYS1.MAN*x*, where *x* is any alphabetic or numeric character (that is, from A through Z, or from 0 through 9). At any one time, a single dataset is the current SMF dataset. SMF uses this dataset, until it becomes full, for recording information.

When the current SMF dataset becomes full, a switch is automatically performed to an empty dataset, which becomes the current dataset, and a message is issued. A message is also issued if no alternate dataset is available.

In response to either message, the operator must dump and clean full datasets so that they can be used again by SMF. Failing to do so can result in losing important system information.

This solution responds to both messages (which means that SMF datasets were switched or that no datasets are available) by initiating a pre-scheduled job in CONTROL-M to dump and clean the full SMF datasets.

Rules

The Copy and Clean SYS1.MAN*x* solution includes the **SMF Dataset Switched, No Empty Dataset Found** rule.

Rules Structure

The following tables describe the structure of the Copy and Clean SYS1.MANx solution rules.

Table 63 SMF Dataset Switched, No Empty Dataset Found Rule Structure (part 1 of 5)

Item	Description
Title	**SMF Dataset Switched, No Empty Dataset Found**
Name	**IEE362A**
Table	**SMF**

Table 63 SMF Dataset Switched, No Empty Dataset Found Rule Structure (part 2 of 5)

Item	Description

Message	Any of the following messages:
	
	`IEE362A SMF ENTER DUMP FOR SYS1.MANx ON volser`
	
	`IEE361I SMF DATA LOST - NO DATASETS AVAILABLE, DATA BEING BUFFERED TIME = hh.mm.ss`
	
	`IEE366I NO SMF DATASETS AVAILABLE - DATA BEING BUFFERED TIME= hh.mm.ss`

Message Description	■ IEE362A – The current SMF dataset has filled up, or a HALT EOD or SWITCH SMF command has been issued. No more SMF records are written to the old SMF dataset, whose name and volser are indicated in the message. ■ IEE361I, IEE366I – The current SMF dataset has filled up and no empty alternate dataset was found.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	%%DUMP_SYS1MAN*x*\ Flag that indicates whether to dump SMF dataset SYS1.MAN*x*. This variable is set for every SMF dataset. Valid values are: ■ YES ■ NO

Table 63 SMF Dataset Switched, No Empty Dataset Found Rule Structure (part 3 of 5)

Item	Description
Rule Logic	The rule is triggered either when SMF dataset SYS1.MANx is switched, or when no empty SMF dataset is found. Either situation requires a user response to initiate a job or started task that executes utility IFASMFDP. This utility dumps and cleans the full SMF datasets. The rule issues command D SMF to obtain SMF dataset information from response message IEE949I and then uses the “percentage-full” value of each dataset to determine which SMF datasets are to be dumped. The current SMF dataset is not dumped by the rule. Flags for each SMF dataset, which indicate individually whether to dump the dataset, are set in Global AutoEdit variables that are referenced by the job JCL. This is done by including the CONTROL-O $GLOBAL member in the job JCL and defining %%LIBSYM and %%MEMSYM AutoEdit control statements. The rule sets a condition to trigger a pre-scheduled job in CONTROL-M. The job executes utility IFASMFDP that dumps SMF datasets. The job must be defined as a cyclic job or cyclic started task (STC). It is triggered by adding the prerequisite condition or date CTO-SMFDUMP-GO 0101. (For more details on the job scheduling definition, see Customization in this table.) Once the rule has been triggered, it is temporarily deactivated by setting its own inverse IN condition. This prevents multiple triggering of the rule caused by messages appearing before the job has finished cleaning the SYS1.MANx datasets. Once the job has successfully executed, condition CTO-IEE362A-HANDLED is deleted to reactivate the rule. The condition that triggered the cyclic job is also deleted upon completion of the job.

Table 63 SMF Dataset Switched, No Empty Dataset Found Rule Structure (part 4 of 5)

Item	Description
Rule Actions	■ Sets a condition to deactivate the rule temporarily (see Rule Logic in this table.) ■ Sets variable %%RESPMSG to IEE949I. ■ Issues operator command DISPLAY SMF. ■ If a response is received, the following actions are performed: – Analyzes response message IEE949I and sets Global variable %%DUMP_SYS1MAN*x* for each SMF dataset to either YES (if it needs dumping) or NO (if it does not need dumping). – Issues a command to instruct CONTROL-O to write the Global variables. – Adds a prerequisite condition to start a job in CONTROL-M to dump the SMF dataset.
Activating the Rule	Once ordered, the rule remains active until one of the messages IEE362A, IEE361I or IEE366I exceeds a predefined threshold. (For more information regarding threshold handling, see "SMF Rule Thresholds" on page 131.) The rule is also temporarily deactivated when it is triggered and reactivated after the copy and clean job finishes OK (see Rule Logic in this table).

Table 63 SMF Dataset Switched, No Empty Dataset Found Rule Structure (part 5 of 5)

Item	Description
Recommended Mode or Category	If a different automatic mechanism to clean SYS1.MAN*x* (for example, SMF Exit IEFU29) is already implemented, the mechanism must be removed before testing this rule. During the testing period the rule must be activated in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	The rule dumps each dataset whose "percentage-full" value is not less than a certain value. Currently, the rule dumps datasets that are at least 90% full. This value must be adapted (in the rule definition) to site requirements. A job scheduling definition and JCL for the SYS1.MAN*x* copy and clean job must be created. The SOLVSCHD and SOLVJCL libraries contain a sample scheduling definition and JCL to copy and clean SYS1.MAN*x*. These samples can be adapted to a site's conventions and requirements. The job must be defined as a cyclic job or cyclic started task (STC) with a MAXWAIT value of 99. It then only needs to be ordered once, but must not be removed manually from the CONTROL-M Active Jobs file. The MAXWAIT value of 99 ensures that the job is never removed from the Active Jobs file by the CONTROL-M New Day procedure. The cyclic job is always ready for submission. It is triggered by adding the prerequisite condition or date CTO-SMFDUMP-GO 0101. When an execution of the job is completed, this condition is deleted. This prevents cyclic re-invoking of the job and ensures that the job is only invoked again if the rule is triggered again. Prerequisite condition or date CTO-IEE362A-HANDLED 0101 must be deleted at time of IPL to make sure the rule is active after system startup (see "SolveWare Initialization" on page 31.)

SMF Alerts

This solution handles various SMF error situations that are not resolved automatically. This solution ensures that the system administrator is notified immediately by CONTROL-O when SMF problems occur, so that the system administrator can locate the source of the error and solve the problem.

Rules

The SMF Alerts solution includes the following rules:

- SMF Dataset Resides on a Non-Direct Access Volume
- I/O Error While Writing on SMF Dataset
- SMF Dataset or SYS1.PARMLIB Cannot Be Opened

Rules Structure

The following tables describe the structures of the SMF Alerts solution rules.

Table 64 SMF Dataset Resides on a Non-Direct Access Volumes (part 1 of 2)Rule Structure

Item	Description
Title	**SMF Dataset Resides on a Non-Direct Access Volumes**
Name	**IEE3631**
Table	**SMF**
Message	`IEE363I SMF ser NOT DIRECT ACCESS`
Message Description	SMF dataset SYS1.MAN*x* resides on a non-direct access device and therefore cannot be used by SMF.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	IN !CTO-IEE363I-THRESH STAT
Global Variables	None.
Rule Logic	The rule is triggered when the SYS1.MAN*x* dataset resides on a non-direct access device. CONTROL-O immediately brings the error to the attention of the system administrator who can then determine the cause of the problem and proceed to solve it.
Rule Actions	Sends a message notifying user U-SYSADMIN of the problem.

Table 64 SMF Dataset Resides on a Non-Direct Access Volumes (part 2 of 2)Rule Structure

Item	Description
Activating the Rule	Once ordered, the rule remains active until message IEE363I exceeds a predefined threshold. (For more information regarding threshold handling, see "SMF Rule Thresholds" on page 131.)
Recommended Mode or Category	This rule must be activated in PROD mode. The SolveWare category for this rule is 1—little or no customization is required before implementation.

Table 65 SI/O Error While Writing on SMF Dataset Rule Structure

Item	Description
Title	**SI/O Error While Writing on SMF Dataset**
Name	**IEE3641**
Table	**SMF**
Message	`IEE364I SMF {LOGICAL | PHYSICAL} I/O ERROR ON SYS1.MAN`*x* `{FEEDBACK CODE =` *`fc | error text`*`}`
Message Description	An I/O error occurred while writing on the SYS1.MAN*x* dataset indicated in the message.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	The rule is triggered when an error occurs while writing on SYS1.MAN*x*. CONTROL-O immediately brings the error to the attention of the system administrator who can then determine the cause of the problem and proceed to solve it.
Rule Actions	Sends a message notifying user U-SYSADMIN of the problem.
Activating the Rule	Once ordered, the rule remains active until message IEE364I exceeds a predefined threshold. (For more information regarding threshold handling, see "SMF Rule Thresholds" on page 131.)
Recommended Mode or Category	This rule must be activated in PROD mode. The SolveWare category for this rule is 1—little or no customization is required before implementation.

Table 66 SMF Dataset or SYS1.PARMLIB Cannot Be Opened Rule Structure

Item	Description
Title	**SMF Dataset or SYS1.PARMLIB Cannot Be Opened**
Name	**IEE3651**
Table	**SMF**
Message	`IEE365I SMF SYS1.{MANx | PARMLIB} NOT OPENED`
Message Description	SMF failed to open SMF dataset SYS1.MAN*x* or SYS1.PARMLIB. No records are written to the SMF dataset.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	The rule is triggered when SYS1.MAN*x* or SYS1.PARMLIB cannot be opened. CONTROL-O immediately brings the error to the attention of the system administrator who can then determine the cause of the problem and proceed to solve it.
Rule Actions	Sends a message notifying user U-SYSADMIN of the problem.
Activating the Rule	Once ordered, the rule remains active until message IEE365I exceeds a predefined threshold. (For more information regarding threshold handling, see “SMF Rule Thresholds” on page 131.)
Recommended Mode or Category	This rule must be activated in PROD mode. The SolveWare category for this rule is 1—little or no customization is required before implementation.

SMF Rule Thresholds

This solution handles message overload – situations in which a message appears on the console more times than is acceptable. If an SMF message appears too often on the system console, threshold rules deactivate the relevant SMF rule until the source of the problem is found and the problem corrected.

For more information regarding threshold handling., see “SolveWare Implementation Considerations” on page 24.

Rules

The SMF Rule Thresholds solution includes the following rules:

- Handling Exceeded SMF Thresholds
- Resetting SMF Rule Threshold Conditions

Rules Structure

The following tables describe the structures of the SMF Rule Thresholds solution rules.

Table 67 Handling Exceeded SMF Thresholds Rule Structure (part 1 of 2)

Item	Description
Title	**Handling Exceeded SMF Thresholds**
Name	**IEE361I**
Table	**SMF**
Message	Any of the following messages: ■ `IEE361I SMF DATA LOST - NO DATASETS AVAILABLE, DATA BEING BUFFERED TIME=hh.mm.ss` ■ `IEE362A SMF ENTER DUMP FOR SYS1.MANx ON volser` ■ `IEE363I SMF ser NOT DIRECT ACCESS` ■ `IEE364I SMF {LOGICAL | PHYSICAL} I/O ERROR ON SYS1.MANx {FEEDBACK CODE = fc | error text}` ■ `IEE365I SMF SYS1.{MANx | PARMLIB} NOT OPENED` ■ `IEE366I NO SMF DATASETS AVAILABLE - DATA BEING BUFFERED TIME=hh.mm.ss`
Message Description	Descriptions for messages handled by this rule are found in the other rule descriptions belonging to SolveWare subject SMF.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.

Table 67 Handling Exceeded SMF Thresholds Rule Structure (part 2 of 2)

Item	Description
Rule Logic	To avoid message overload situations, this rule deactivates SMF rules whose messages have exceeded a pre-determined number of appearances in a period of time. These threshold values are defined for every message included in this rule in the threshold parameters APPEARED ### TIMES IN #### MINUTES. To synchronize threshold handling correctly, this rule is assigned a higher PRIORITY value than the message rules that it monitors, and has a CONTINUE SEARCH value of Y (Yes). Deactivation of an SMF rule is achieved by adding the appropriate (inverse) IN prerequisite condition for the rule. To reactivate a deactivated rule, the threshold conditions must be deleted. This can be done either manually or automatically by CONTROL-O (see the following section). Threshold conditions must be specified in the IGNORE list of the CONTROL-M CONTDAY procedure (see "SolveWare Implementation Considerations" on page 24).
Rule Actions	■ Notifies user U-SYSADMIN that the message that exceeded its threshold is no longer handled by CONTROL-O. ■ Sets the appropriate IN condition to deactivate the rule. The format of this condition is CTO-*msgid*-THRESH, where *msgid* is the message ID of the specific message.
Activating the Rule	Once scheduled, the rule is triggered whenever one of the above messages exceeds its threshold.
Recommended Mode or Category	During the testing period, the rule must be activated in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Review each SMF message monitored by this rule and determine appropriate threshold values. For each message included in this rule, adapt to site requirements APPEARED ### TIMES IN #### MINUTES values, which specify a number of appearances in a time period.

Table 68 Resetting SMF Rule Threshold Conditions Rule Structure

Item	Description
Title	**Resetting SMF Rule Threshold Conditions**
Name	**RESSMF**
Table	**SMF**
Event	RESSMF
Event Description	This Event rule deletes all threshold conditions for all SMF rules.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	Using the INTERVAL parameter, this rule periodically deletes the threshold conditions of all SMF rules in order to reactivate SMF rules that exceeded their thresholds. (For further information regarding the resetting of threshold conditions, see Customization in this table.)
Rule Actions	■ Deletes condition or date CTO-IEE361I-THRESH STAT ■ Deletes condition or date CTO-IEE362A-THRESH STAT ■ Deletes condition or date CTO-IEE363I-THRESH STAT ■ Deletes condition or date CTO-IEE364I-THRESH STAT ■ Deletes condition or date CTO-IEE365I-THRESH STAT ■ Deletes condition or date CTO-IEE366I-THRESH STAT
Activating the Rule	Once scheduled, the rule is triggered periodically according to the INTERVAL parameter specification.
Recommended Mode or Category	During the testing period, the rule must be activated in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Adapt the INTERVAL parameter to site requirements: If you use the INTERVAL parameter without the TIME FROM parameter, the threshold conditions are deleted when the rule is ordered.
	Note: Threshold conditions must also be deleted at time of IPL. For more information, see the section on initializing SolveWare in Chapter 2, “STARTSYS.” Threshold conditions must be specified in the IGNORE list of the CONTROL-M CONTDAY procedure. For details, see the section on SolveWare implementation considerations in Chapter 1, “Introduction.”

DUMPDS

DUMPDS solutions are designed to automate management of system dump datasets. Various components of the MVS environment produce dumps that include diagnostic information regarding a problem. These dumps are kept in the dump dataset SYS1.DUMP*xx*.

A common way of managing the dump files is to ensure that there are always several empty dump files to which new dumps can be written. This solution implements this method of work.

NOTE

All messages (DO SHOUT actions) in the DUMPDS rules are sent to an INCONTROL user named U-SYSADMIN. A user with this name must be defined in the IOA Dynamic Destination table (CTMDEST). For more information, see the Dynamic Destination Table chapter of the *INCONTROL for z/OS Administrator Guide*.

In many cases, rules definitions make use of the inverse IN Condition feature. This feature activates rules only if the specified IN conditions are not set. For more information about using inverse IN conditions, see the section on SolveWare implementation considerations in Chapter 1, “Introduction.”

Solutions Provided

SolveWare subject DUMPDS contains the following solutions:

- Copy and Clean SYS1.DUMP*xx*

  For situations where an insufficient number of SYS1.DUMP files are available. A copy and clean job is automatically sent to prevent loss of system information.

- DUMPDS Rule Thresholds

  Handles exceeded thresholds of all other DUMPDS rules.

Copy and Clean SYS1.DUMP*xx*

In an MVS environment, a predefined number of dump datasets is defined and allocated. These files are named SYS1.DUMPxx. Whenever any system component produces a dump, it is directed to one of the empty (available) dump files.

If all dump files are full, the current dump cannot be produced and important information may be lost.

This solution is designed to ensure that a pre-specified number of empty dump datasets is always available for new dumps. The solution responds to MVS messages that indicate that no dump datasets are available, and after a dump has been produced, it checks that there is still a sufficient number of empty dump datasets.

Rules

The Copy and Clean SYS1.DUMPxx solution includes the **Complete or Partial Dump Produced, All Dump Datasets Full** rule.

Rules Structure

The following tables describe the structures of the **Complete or Partial Dump Produced, All Dump Datasets Full** solution rules.

Table 69 Complete or Partial Dump Produced, All Dump Datasets Are Full Rule Structure (part 1 of 5)

Item	Description
Title	**Complete or Partial Dump Produced, All Dump Datasets Full**
Name	**IEA911E**
Table	**DUMPDS**

Table 69 Complete or Partial Dump Produced, All Dump Datasets Are Full Rule Structure (part 2 of 5)

Item	Description

Message	Any of the following messages:
	
	`IEA911E { COMPLETE | PARTIAL } DUMP ON SYS1.DUMP`*xx* `FOR ASID` *asn*
	
	`IEA994A ALL DUMP DATASETS ARE FULL AND NO SVC DUMPS CAN BE PRODUCED`
	
	`IEA994E ALL ALLOCATED SYS1.DUMP DATASETS ARE FULL`
	
	`IEE711I { SYSTEM UNABLE TO DUMP | SYSTEM DUMP NOT PRODUCED` *reason-text* `}`

Message Description	■ IEA911E – An SVC dump was written to a SYS1.DUMP dataset. ■ IEA994A, IEA994E – All SYS1.DUMP datasets are full. The current dump, and subsequent dumps, cannot be produced. ■ IEE711I – An SVC dump cannot be produced.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	■ %%DUMP_AVAIL The number of empty dump datasets that must be available for new dumps at any time. ■ %%CLEAR_DUMP*xx* Flag indicating whether to clear dump dataset SYS1.DUMP*xx*. This variable is set for every dump dataset in the system. Valid values are: – YES – NO

Table 69 Complete or Partial Dump Produced, All Dump Datasets Are Full Rule Structure (part 3 of 5)

Item	Description
Rule Logic	This rule is triggered when a dump is produced, or when no empty dump dataset is found. Either situation normally requires a user response. The user must check if a sufficient number of empty dump datasets is available, and if not, the user must clear dump datasets using command DUMPDS, or utility IEBGENER, or using IPCS. The rule issues command DISPLAY DUMP to obtain SYS1.DUMP dataset information that it uses to determine the number of dump datasets currently available. The rule then checks if this number is sufficient according to the value specified by the user in Global AutoEdit variable %%DUMP_AVAIL, which that reflects the number of empty dump datasets that are available for new dumps at any time. If enough datasets are empty, the rule ends and no actions are performed. If there is a need to empty dump files, then for each SYS1.DUMP dataset, the rule sets a flag indicating whether to clear the dataset. The flags are set in Global AutoEdit variables referenced by the job JCL. The rule adds prerequisite condition or date CTO-CLRDUMP-GO 0101, which triggers a pre-scheduled job in CONTROL-M to clear the dump datasets. (For further details on the job scheduling definition, see Customization in this table.) Once the rule has been triggered, it is temporarily deactivated by setting its own inverse IN condition. This prevents multiple triggering of the rule caused by additional appearances of the above messages before the job has finished cleaning the specified SYS1.DUMPxx datasets. Once the job has successfully executed, condition CTO-IEA911E-HANDLED is deleted to reactivate the rule. Condition CTO-CLRDUMP-GO, which triggered the pre-scheduled job, is deleted upon completion of the job.

Table 69 Complete or Partial Dump Produced, All Dump Datasets Are Full Rule Structure (part 4 of 5)

Item	Description
Rule Actions	■ Sets Global AutoEdit variable %%DUMP_AVAIL to 1. This variable reflects the number of empty dump datasets that must be available at any time (see Rule Logic in this table). The default value of 1 can be changed by the user—see Customization in this table. ■ Sets condition CTO-IEA911E-HANDLED to deactivate the rule temporarily (see Rule Logic in this table). ■ Sets variable %%RESPMSG to IEE852I. ■ Issues operator command DISPLAY DUMP. ■ Analyzes response message IEE852I to determine the number of currently available dump datasets. ■ Sets Global AutoEdit variable %%CLEAR_DUMP*xx* to YES or NO to indicate which datasets to empty. ■ Issues a command instructing CONTROL-O to write Global AutoEdit variables. ■ Sets prerequisite condition CTO-CLRDUMP-GO to start a job in CONTROL-M to clear flagged dump datasets.
Activating the Rule	Once ordered, the rule remains active until one of the messages IEA911E, IEA994A, IEA994E, or IEE711I exceeds a pre-defined threshold. (For more information regarding threshold handling, see "DUMPDS Rule Thresholds" on page 140.) The rule is also temporarily deactivated when it is triggered and reactivated after the copy and / or clean job finishes OK (see Rule Logic in this table).
Recommended Mode or Category	During the testing period the rule must be activated in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different automatic mechanism to clean SYS1.DUMP*xx* is already implemented, the mechanism must be removed before testing this rule. A job scheduling definition and JCL member for the SYS1.DUMPxx copy and or clean job must be created. SOLVSCHD and SOLVJCL Libraries contain a sample scheduling definition and JCL to copy and or clean SYS1.DUMP*xx*. This sample can be adapted to site conventions and requirements. The supplied sample job executes program utility IPCS, which prints and clears the dump datasets. This job can be modified to use a different method to clear the dump files such as the DUMPDS command.

Table 69 Complete or Partial Dump Produced, All Dump Datasets Are Full Rule Structure (part 5 of 5)

Item	Description
Customization (continued)	The supplied sample job includes %%LIBSYM and %%MEMSYM AutoEdit control statements, which reference the CONTROL-O $GLOBAL member. SYS1.DUMP flags (AutoEdit variables that indicate whether to clear specific datasets) must be included in this member. The job must be defined as a cyclic job or cyclic started task (STC) with a MAXWAIT value of 99. It then only needs to be ordered once, but must not be removed manually from the CONTROL-M Active Jobs file. The MAXWAIT value of 99 ensures that the job is never removed from the Active Jobs file by the CONTROL-M New Day procedure. The cyclic job is always ready for submission. It is triggered by adding the prerequisite condition or date CTO-CLRDUMP-GO 0101. When an execution of the job is completed, this condition is deleted. This prevents cyclic re-invoking of the job and ensures that the job is only invoked again if the rule is triggered again. If there is a need to empty dump files, then for each SYS1.DUMP dataset, the rule sets a flag indicating whether to clear the dataset. The flags are set in Global AutoEdit variables referenced by the job JCL. This is done by including the CONTROL-O $GLOBAL member in the job JCL and defining %%LIBSYM and %%MEMSYM AutoEdit control statements. In the rule definition, the default value supplied for Global AutoEdit variable %%DUMP_AVAIL is 1, meaning that at least one dump dataset must be left empty at all times. This value must be adapted to site requirements. Prerequisite condition or date CTO-IEA911E-HANDLED STAT must be deleted at time of IPL to make sure the rule is active after system startup. For more information, see the section on initializing SolveWare in Chapter 2, “STARTSYS.”

DUMPDS Rule Thresholds

This solution handles message overload – situations in which a message appears on the console more times than is desirable. If a SYS1.DUMP related message appears too often on the system console, threshold rules deactivate the relevant DUMPDS rule until the source of the problem is found and the problem corrected.

For more information on threshold handling, see the section about SolveWare implementation considerations in Chapter 1, “Introduction.”

Rules

The DUMPDS Rule Thresholds solution includes the following rules:

- Handling Exceeded DUMPDS Thresholds
- Resetting DUMPDS Rule Threshold Conditions

Rules Structure

The following tables describe the structures of the DUMPDS Rule Thresholds solution rules.

Table 70 Handling Exceeded DUMPDS Thresholds Rule Structure (part 1 of 2)

Item	Description
Title	**Handling Exceeded DUMPDS Thresholds**
Name	**IEA911E**
Table	**DUMPDS**

Message	Any of the following messages:
	
	`IEA911E { COMPLETE | PARTIAL } DUMP ON SYS1.DUMP`*`xx`*
	`FOR ASID `*`asn`*
	
	`IEA994A ALL DUMP DATASETS ARE FULL AND NO SVC DUMPS CAN BE PRODUCED`
	
	`IEA994E ALL ALLOCATED SYS1.DUMP DATASETS ARE FULL`
	
	`IEE711I { SYSTEM UNABLE TO DUMP | SYSTEM DUMP NOT PRODUCED `*`reason-text`*` }`

Message Description	All messages handled by rules comprising the DUMPDS solutions.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.

Table 70 Handling Exceeded DUMPDS Thresholds Rule Structure (part 2 of 2)

Item	Description
Rule Logic	To avoid message overload situations, this rule deactivates DUMPDS rules whose messages have exceeded a pre-determined number of appearances in a period of time. These threshold values are defined for every message included in this rule in the threshold parameters APPEARED ### TIMES IN #### MINUTES. To synchronize threshold-handling correctly, this rule is assigned a higher PRIORITY value than the message rules that it monitors and has a CONTINUE SEARCH value of Y (Yes). Deactivation of a DUMPDS rule is achieved by adding the appropriate (inverse) IN prerequisite condition that is defined for the rule. To reactivate a deactivated rule, the threshold conditions must be deleted. This can be done either manually or automatically by CONTROL-O. Threshold conditions must be specified in the IGNORE list of the CONTROL-M CONTDAY procedure (see "SolveWare Implementation Considerations" on page 24).
Rule Actions	■ Notifies user U-SYSADMIN that the message that has exceeded its threshold is no longer handled by CONTROL-O. ■ Sets the appropriate IN condition to deactivate the rule. The format of this condition is CTO-*msgid*-THRESH, where *msgid* is the message ID of the specific message.
Activating the Rule	Once scheduled, the rule is triggered whenever one of the above messages exceeds its threshold.
Recommended Mode or Category	During the testing period, the rule must be activated in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Review each DUMPDS message monitored by this rule and determine threshold parameters. Take into consideration the total number of dump datasets in your site, the frequency of producing dumps, and so on. For every message included in this rule, adapt APPEARED ### TIMES IN #### MINUTES values, which specify a number of appearances in a time period, to site requirements.

Table 71 Resetting DUMPDS Rule Threshold Conditions Rule Structure

Item	Description
Title	**Resetting DUMPDS Rule Threshold Conditions**
Name	**RESDUMP**
Table	**DUMPDS**
Event	RESDUMP
Event Description	This Event rule deletes all threshold conditions for all DUMPDS rules.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	Using the parameters INTERVAL and TIME FROM, this rule periodically deletes the threshold conditions of all DUMPDS rules in order to reactivate DUMPDS rules that exceeded their thresholds. (For more information about resetting threshold conditions, see Customization in this table.)
Rule Actions	■ Deletes condition or date CTO-IEA911E-THRESH STAT ■ Deletes condition or date CTO-IEA994A-THRESH STAT ■ Deletes condition or date CTO-IEA994E-THRESH STAT ■ Deletes condition or date CTO-IEE711I-THRESH STAT
Activating the Rule	Once scheduled, the rule is triggered periodically according to the INTERVAL parameter specification.
Recommended Mode or Category	During the testing period, the rule must be activated in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Adapt the INTERVAL parameter to site requirements: If you use parameter INTERVAL without parameter TIME FROM, the threshold conditions are deleted when the rule is ordered.
	Note: Threshold conditions must also be deleted at time of IPL (for details, see the section on initializing SolveWare in Chapter 2, “STARTSYS.” Threshold conditions must be specified in the IGNORE list of the CONTROL-M CONTDAY procedure. For more information, see the section about SolveWare implementation considerations in Chapter 1, “Introduction.”

CONSOLES

Solutions in SolveWare subject CONSOLES are designed to automate WTO (Write to Operator) message buffer management.

Solutions Provided

SolveWare subject CONSOLES contains the WTO Buffer Shortage solution, which relieves WTO buffer shortages.

WTO Buffer Shortage

This solution automates the process of relieving WTO buffer shortage situations. WTO buffer shortage situations usually occur

- when a console is not placed in Roll mode.
- when the console is filled with unrollable messages kept by WTOR or MRF (Message Retention Facility) messages.
- when the console device cannot handle the messages at the speed at which messages are sent to the console.

WTO buffer shortages generally lead to overall performance degradation.

When 80 percent of the WTO buffers are full, MVS issues a message indicating this situation. If steps to alleviate the situation are not taken and all WTO buffers become full, the console address space and any address space issuing a WTO, enters a Wait state until WTO buffers are made available.

This solution relieves WTO buffer shortages by either rerouting to the hardcopy log those messages that are queued to problematic consoles, or by deactivating the console.

NOTE

All messages (DO SHOUT actions) in the CONSOLES rules are sent to an INCONTROL user named U-SYSADMIN. A user with this name must be defined in the IOA Dynamic Destination table (CTMDEST). For more information, see the Dynamic Destination Table chapter of the *INCONTROL for z/OS Administrator Guide*.

Rules

The WTO Buffer Shortage solution includes the **Relieve WTO Buffer Shortage** rule.

Rules Structure

The following tables describe the structures of the WTO Buffer Shortage solution rules.

Table 72 Relieve WTO Buffer Shortage Rule Structure

Item	Description
Title	**Relieve WTO Buffer Shortage**
Name	**IEA405E**
Table	**CONSOLES**
Message	`IEA405E WTO BUFFER SHORTAGE - 80% FULL`
Message Description	The number of WQEs (WTO Queue Elements) in use for console messages has reached 80 percent of its maximum capacity.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%CSCTR_*consoleid* Number of times the messages for the specific console id were rerouted by the rule to the hardcopy log.
Rule Logic	This message rule is triggered when 80 percent of the WTO buffers are full. The rule issues MVS command DISPLAY CONSOLES in command-response mode. The response messages are analyzed in order to locate any console that has more then 50 messages queued to it. For those consoles, the messages are rerouted to the hardcopy log. If a console is still accumulating buffers after 10 rerouting actions, the console is deactivated.

Table 72 Relieve WTO Buffer Shortage Rule Structure

Item	Description
Rule Actions	■ Issues MVS command DISPLAY CONSOLES in Command-Response mode. ■ Identifies those consoles that have more than 50 buffers queued to them. ■ If SYSLOG is active, the rule reroutes the consoles' buffers to hardcopy using MVS command K Q,R=HC,L=*consoleid*. ■ Consoles still accumulating buffers after 10 rerouting actions are deactivated using MVS command V *unitname*,ONLINE – where *unitname* is the MVS address of the console. ■ Issues appropriate messages to the console and to the IOA Log.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, the rule must be activated in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Due to the IBM console restructuring process that was introduced in z/OS version 1.4, there may be changes in the format of the response messages for the DISPLAY CONSOLES command. The current format is that of z/OS version 1.4.

Chapter 10

TSO

Solutions in SolveWare subject TSO are designed to automate various management aspects of TSO.

Solutions Provided

SolveWare subject TSO contains the Maintain Messages in the TSO Broadcast Dataset solution. This solution provides an easy way to define and save messages in the notices section of dataset SYS1.BRODCAST. These notice messages are displayed when a user logs onto TSO.

Maintain Messages in the TSO Broadcast Dataset

TSO supports notice *messages* – *messages* displayed to all users during logon to TSO. These messages are stored in the notices section of dataset SYS1.BRODCAST. This dataset is a direct access dataset and cannot be edited. The only way to add or delete broadcast messages is by using MVS SEND commands – one command for each line to be added or deleted.

This solution provides an easy method by which the system administrator can maintain messages in the notices section of SYS1.BRODCAST. The system administrator simply edits the messages in a sequential dataset and issues a special user- defined operator command that triggers the Command rule described below.

This Command rule clears the notices section of the broadcast dataset and initiates a KOA script to read the dataset containing the message. Each message is passed to another rule that issues a corresponding MVS SEND operator command in order to save the message in the notices section of SYS1.BRODCAST.

Rules

The Maintain Messages in the TSO Broadcast Dataset solution includes the following rules:

- Operator Command to Clear or Set Broadcast Messages
- Set a Given Broadcast Message in SYS1.BRODCAST

Rules Structure

The following tables describe the structures of the Maintain Messages in the TSO Broadcast Dataset solution rules, as well as the following KOA script:

- Read Broadcast Messages From a File and Trigger a Rule to Save the Messages

Table 73 Command to Clear or Set Broadcast Messages Rule Structure (part 1 of 2)

Item	Description
Title	**Command to Clear or Set Broadcast Messages**
Name	**BRODCAST**
Table	**TSO**

Command	One of the following user-defined commands:
	
	`BRODCAST CLEAR`
	
	`BRODCAST SET [`*dsn*`]`
	
	`BRODCAST [HELP]`

Command Description	BRODCAST CLEAR
	
	Deletes all the messages from the notices section of SYS1.BRODCAST
	
	`BRODCAST SET [`*dsn*`]`
	
	Saves all messages from dataset dsn in the notices section of SYS1.BRODCAST. If dsn is not specified, a default dataset name is used. All previous notice messages are deleted.
	
	`BRODCAST [HELP]`
	
	Displays the syntax of command BRODCAST on the console.

Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Table 73 Command to Clear or Set Broadcast Messages Rule Structure (part 2 of 2)

Item	Description
Global Variables	None.
Rule Logic	If the command parameter is HELP, the command syntax is displayed on the console. If the command parameter is CLEAR or SET, the rule issues MVS command SEND LIST in command-response mode. For each response message, the rule extracts the message number and deletes the broadcast message by issuing MVS command SEND *msgno*,DELETE. If the command parameter is SET, the rule starts KOA script BRODCAST, with either the default or the specified dataset name parameter. (The KOA script reads messages from the specified dataset, and for each message, triggers an additional rule to set the broadcast message.) After the KOA script finishes, the rule issues a message indicating whether the broadcast dataset has been updated. For any other command parameter, the rule issues an error message.
Rule Actions	■ If the command parameter is HELP, displays command syntax on the console. ■ If the command parameter is CLEAR or SET, the rule issues MVS command SEND LIST in command-response mode. For each response message, the rule extracts the message number and deletes the broadcast message by issuing MVS command SEND msgno,DELETE. ■ If the command parameter is SET, starts KOA script BRODCAST, with either the default or the specified dataset name parameter. (The KOA script reads messages from the specified dataset, and for each message, triggers an additional rule to set the broadcast message.) After the KOA script finishes, the rule issues a message indicating whether the broadcast dataset has been updated. ■ For any other command parameter, the rule issues an error message.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O. The rule is triggered by each BRODCAST operator command.
Recommended Mode or Category	During the testing period, the rule activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	The default name of the dataset in which the broadcast messages were edited is specified in the first DO SET statement of the rule (variable name %%BRODCAST_DSN). Set this variable according to site requirements.

Table 74 Set a Given Broadcast Message in SYS1.BRODCAST Rule Structure

Item	Description
Title	**Set a Given Broadcast Message in SYS1.BRODCAST**
Name	**CTM2821**
Table	**TSO**
Message	`CTM282I BR=`*`text-of-brodcast-message`*
Message Description	The message is issued by KOA script BRODCAST and contains the text of the message to be saved in the broadcast dataset.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	Extract the text of the message and issue MVS command SEND '*text*',ALL,SAVE.
Rule Actions	Extracts the text of the message. Issues MVS command SEND '*text*',ALL,SAVE.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1—little or no customization is required before implementation.

Table 75 Read Broadcast Messages from a File and Trigger a Rule to Save the Messages Script Structure (part 1 of 2)

Item	Description
Title	Read Broadcast Messages from a File and Trigger a Rule to Save the Messages
Name	BRODCAST
KOA Script Description	This KOA script is invoked by rule BRODCAST. The rule passes to the script a parameter that identifies the dataset containing the text of the broadcast messages. The KOA script allocates and opens the dataset. Each record is read by the script and the text is issued as a WTO message by using a SHOUT command. This triggers rule CTM282I, which issues the corresponding MVS SEND command to save the messages. The script then closes and frees the dataset.
Activating the KOA Script	The KOA script is activated by rule BRODCAST in this solution.
Parameters	%A1 Name of the dataset that contains the messages. It must be a sequential dataset.

Table 75 Read Broadcast Messages from a File and Trigger a Rule to Save the Messages Script Structure (part 2 of 2)

Global Variables	None.
KOA Script Logic	The KOA script allocates and opens the dataset passed as a parameter. Each record is read and the text is issued as a WTO message using a SHOUT command. This triggers rule CTM282I, which issues the corresponding MVS SEND command to save the messages. The dataset is finally closed and freed. No logon to a VTAM application is performed by this script.
Recommended Mode or Category	During the testing period, the script can run with the TRACE ON option. When you are satisfied with the results, change this statement to TRACE OFF to avoid unnecessary tracing. The SolveWare category for this KOA script is 1—no customization (or minimal) is required before implementation.

Chapter 11

CICS

Solutions in the SolveWare subject CICS are designed to automate several management aspects of the Customer Information Control System (CICS).

NOTE

All messages (DO SHOUT actions) in the CICS rules are sent to an INCONTROL user named U-SYSCICS. A user with this name must be defined in the IOA Dynamic Destination table (CTMDEST). For more information, see the Dynamic Destination Table chapter of the *INCONTROL for z/OS Administrator Guide*.

Solutions Provided

SolveWare subject CICS contains the following solutions:

- CICS Startup and Termination

  Monitors the CICS startup and termination processes, intercepts relevant messages and updates the appropriate IOA prerequisite conditions.

- Copy and Clean CICS Journal

  Handles situations where the CICS journal datasets become full. A copy and clean job is submitted automatically to prevent loss of information.

CICS Startup and Termination

This solution handles CICS startup and termination.

A connection between CICS and INCONTROL products is established by defining IOA prerequisite conditions that reflect CICS's status (up or down). Specifying these conditions as prerequisite IN conditions in IOA definitions makes processing dependent upon CICS being up or down.

NOTE

While startup or termination is in process—but not yet completed—the IOA environment considers CICS as both "not up" and "not down." Therefore, CICS not being up is not necessarily the same as a CICS down status and vice versa.

IOA CICS prerequisite conditions must also be updated at time of IPL to indicate that the various CICS subsystems are down. For more information, see the section on initializing SolveWare in Chapter 2, "STARTSYS."

Rules

The CICS Startup and Termination solution includes the following rules:

- CICS Initialization Started
- CICS Initialization Completed
- CICS Termination Started
- CICS Termination Completed
- CICS Severe Termination (JES2 Only)
- Disable S CICS If CICS Is Active
- Startup Continuation Permission

Rules Structure

The following tables describe the structures of the CICS Startup and Termination solution rules.

Table 76 CICS Initialization Started Rule Structure (part 1 of 2)

Item	Description
Title	**CICS Initialization Started**
Name	**IEF403I**

Table 76 CICS Initialization Started Rule Structure (part 2 of 2)

Item	Description
Table	**CICS**
Message	`IEF403I jjj STARTED {- TIME=hh.mm.ss}` when the message is issued from job CICS*.
Message Description	Job *jjj* starts.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	As soon as CICS is started, the IOA environment is informed that CICS is no longer down.
Rule Actions	Deletes condition or date CTO-*cicsname*-DOWN 0101 (where *cicsname* is the name of the CICS job or started task).
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls CICS conditions (utility IOACND, CONTROL-M Event Manager, and so on.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition deleted by this rule to match your previous definition, if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that CICS is down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule.

Table 77 CICS Initialization Completed Rule Structure (part 1 of 2)

Item	Description
Title	**CICS Initialization Completed**
Name	**DFH*1517**
Table	**CICS**
Message	`DFHSI1517 aplid CONTROL IS BEING GIVEN TO CICS`

Table 77 CICS Initialization Completed Rule Structure (part 2 of 2)

Item	Description
Message Description	CICS has completed initialization and is up.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	When CICS is completely up, the IOA environment is informed that CICS is up.
Rule Actions	Adds condition or date CTO-*cicsname*-UP STAT (where *cicsname* is the name of the CICS job or started task).
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls CICS conditions (utility IOACND, CONTROL-M Event Manager, and so on.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition added by this rule to match your previous definition, if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that CICS is up. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule.

Table 78 CICIS Termination Started Rule Structure (part 1 of 3)

Item	Description
Title	**CICIS Termination Started**
Name	**DFH*1071**
Table	**CICS**

Table 78 CICIS Termination Started Rule Structure (part 2 of 3)

Item	Description
Message	One of the following messages: `DFH1701 C.I.C.S IS BEING TERMINATED BY OPERATOR` *`opid`* `AT TERMINAL` *`termid`* `DFHTM1701 {C.I.C.S. | CICS/ESA} IS BEING TERMINATED BY OPERATOR` *`opid`* `AT TERMINAL` *`termid`* `DFHTM1703` *`applid`* `PRODUCT IS BEING TERMINATED BY USERID` *`userid`* `IN TRANSACTION` *`tranid`* `{AT NETNAME | AT TERMINAL}` *`terminal`*. `DFHTM1715` *`applid`* `PRODUCT IS BEING QUIESCED BY USERID` *`userid in`* `TRANSACTION` *`tranid`* `{AT NETNAME | AT TERMINAL}` *`terminal`* `IEF404I` *`jjj`*`-ENDED(- TIME=`*`hh.mm.ss`*`)` when the message is issued from job CICS*
Message Description	■ DFHTM1701, DFH1701, DFHTM1703, and DFHTM1715 – Operator *opid* issued a CICS shutdown request from terminal *termid*. ■ IEF404I – Job *jjj* ended.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	When CICS termination has started, the IOA environment is informed that CICS is no longer up.
Rule Actions	Deletes condition or date CTO-*cicsname*-UP STAT, where *cicsname* is the name of the CICS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.

Table 78 CICIS Termination Started Rule Structure (part 3 of 3)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls CICS conditions (utility IOACND, CONTROL-M Event Manager, and so on.) is already implemented, the mechanism must be removed before implementing this rule. If the IOA environment already contains a prerequisite condition indicating that CICS is up (for example, if such a condition is already referenced by a CONTROL-M job scheduling definition), and if the prerequisite condition has a different name than the prerequisite condition indicated in this rule, then change the name of the prerequisite condition deleted by this rule to match your previous definition.

Table 79 CICS Termination Completed Rule Structure (part 1 of 2)

Item	Description
Title	**CICS Termination Completed**
Name	**DFH*1799**
Table	**CICS**

Message	One of the following messages:
	
	`DFH1799/DFHKE1799 aplid TERMINATION OF CICESA IS COMPLETE`
	
	`IEF404I jjj ENDED{ - TIME=hh.mm.ss}`
	
	when the message is issued from job CICS*.

Message Description	■ DFH1799 / DFHKE1799 – CICS has completed termination. ■ IEF404I – Job *jjj* ended.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	When CICS termination is completed, the IOA environment is informed that CICS is down.
Rule Actions	Adds condition or date CTO-*cicsname*-DOWN STAT (where *cicsname* is the name of the CICS job or started task).

Table 79 CICS Termination Completed Rule Structure (part 2 of 2)

Item	Description
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls CICS conditions (utility IOACND, CONTROL-M Event Manager, and so on.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition added by this rule to match your previous definition, if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that CICS is down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule.

Table 80 CICS Severe Termination Rule Structure (part 1 of 2)

Item	Description
Title	**CICS Severe Termination**
Name	**$HASP310**
Table	**CICS**
Message	`$HASP310 jjj TERMINATED AT END OF MEMORY`
Message Description	Abend of job *jjj* was so severe that no Recovery Termination Manager (RTM) processing occurred.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	For severe CICS abends, the IOA environment is informed that CICS is no longer up and that CICS is down.

Table 80 CICS Severe Termination Rule Structure (part 2 of 2)

Item	Description
Rule Actions	■ Adds condition or date CTO-*cicsname*-DOWN STAT (where *cicsname* is the name of the CICS job or started task that abended). ■ Deletes condition or date CTO-*cicsname*-UP STAT (where *cicsname* is the name of the CICS job or started task that abended).
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls CICS conditions (utility IOACND, CONTROL-M Event Manager, and so on.) is already implemented, the mechanism must be removed before implementing this rule. Change the names of the prerequisite conditions added or deleted by this rule to match your previous definitions, if both of the following are true: The IOA environment already contains prerequisite conditions indicating that CICS is up or down. For example, if such conditions are already referenced by CONTROL-M job scheduling definitions. The prerequisite conditions have different names than the ones indicated in this rule.
Note: This rule applies only at JES2 sites.	

Table 81 Suppress S CICS if CICS Is Active Rule Structure (part 1 of 2)

Item	Description
Title	**Suppress S CICS if CICS Is Active**
Name	**S CICS**
Table	**CICS**
Command	`S CICS`
Command Description	This command starts CICS (the rule suppresses the command).
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	IN !CTO-*cicsname*-DOWN STAT

Table 81 Suppress S CICS if CICS Is Active Rule Structure (part 2 of 2)

Item	Description
Global Variables	None.
Rule Logic	If command S CICS is issued when CICS is already up, the new started task is immediately shut down and normal termination messages are issued. These messages, however, give the appearance that CICS has gone down and therefore may incorrectly trigger rules. This rule prevents this situation by suppressing the S CICS command if CICS is already up. In this way, issuance of the termination messages is prevented and rules are not incorrectly triggered. This rule serves as a sample rule. A similar rule must be created for each CICS environment in the site.
Rule Actions	■ Suppresses the command. ■ Sends a message to user U-SHFTOPER indicating that command S CICS was suppressed.
Activating the Rule	Once scheduled, the rule becomes active when prerequisite condition or date CTO-*cicsname*-DOWN STAT is deleted (where *cicsname* is the name of the CICS address space).
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Change the name of the prerequisite condition indicated in this rule to match your previous definition, if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that CICS is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. The prerequisite condition has a different name than the one indicated in this rule. ■ The rule must be duplicated for each CICS in the site. To adapt each rule, the CICS name in the rule's ON statement and IN condition must be changed.

Table 82 Startup Continuation Permission Rule Structure (part 1 of 2)

Item	Description
Title	**Startup Continuation Permission**
Name	**DFH*1588***
Table	**CICS**

Table 82 Startup Continuation Permission Rule Structure (part 2 of 2)

Item	Description
Message	One of the following messages: `DFH1588/DFHSI1588D IS STARTUP TO BE CONTINUED? REPLY GO OR CANCEL`
Message Description	CICS is requesting permission to proceed after an error has occurred during initialization, or after an emergency restart has completed.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	The rule is triggered after an error has occurred during CICS initialization, or after an emergency restart has completed. CICS issues message DFH1588 and waits for the operator to reply whether startup is to be continued or cancelled. The rule replies to the message with GO, causing the startup operation to continue.
Rule Actions	Replies to the message with GO.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, the rule must be activated in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Automatic reply to the message may be desirable only for certain CICS environments. For example, test environment only. If so, change the JNAME field in the ON statement of the rule definition to intercept messages only from the CICS address spaces of those environments.

Copy and Clean CICS Journal

CICS journals are special-purpose sequential datasets that are accessed at real-time CICS execution by user tasks. Information written to CICS journals can be used subsequently for problem determination, database update log, or as a record of transactions passing through the system.

Each journal is identified by a number ranging from 2 through 99, inclusive, and is defined in the Journal Control Table (JCT). Journal number 1 is reserved for the system log journal.

When a journal dataset resides on disk and is defined in the Journal Control Table (JCT) with JOUROPT=PAUSE, message DFH4583 is issued if the dataset becomes full. CICS then waits for a reply from the operator indicating that the dataset can be used again (meaning, either it has been copied or a copy is not desired).

A correct reply must be received before CICS attempts to access the dataset for output again. Otherwise, all tasks using the specified journal are delayed until the reply is received.

This solution provides two options for handling full journal datasets:

- For a test CICS environment, the rule immediately replies to the message so that CICS reopens the journal dataset for output.
- For a production CICS environment, the provided rules automatically trigger a copy and clean job in CONTROL- M to dump the relevant dataset and reply to the message upon successful completion of the job.

Rules

The Copy and Clean CICS Journal solution includes the following rules:

- CICS Journal Full—Test Environment
- CICS Journal Full—Production Environment
- CICS Journal Copied—Production Environment

Rules Structure

The following tables describe the structures of the Copy and Clean CICS Journal solution rules.

Table 83 CICS Journal Full—Test Environment Rule Structure

Item	Description
Title	**CICS Journal Full—Test Environment**
Name	**DFH*4583**
Table	**CICS**
Message	DFH4583/DFHJC4583 *aplid* CICS {SYSTEM LOG | JOURNAL *nn*} {PRIMARY | SECONDARY | EMERGENCY} DATASET (DDNAME=DFHJ*nnx*) READY TO BE COPIED. REPLY *'Ynnx'* WHEN COPIED. and the job that issued the message is CICST*.
Message Description	The CICS journal dataset indicated in the message has filled up and the journal was specified with the PAUSE option.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	The rule is triggered when a CICS journal dataset becomes full and that journal is specified with the PAUSE option. The message waits for a system operator reply indicating that the dataset can be used again by CICS. A correct reply must be received before CICS attempts to access the dataset for output again. Otherwise, all tasks using the specified journal are delayed until the reply is received. Assuming that journal datasets of the test CICS environment do not need to be copied, the rule immediately replies to the message so that CICS can use the journal dataset again.
Rule Actions	Determines the reply string according to message text. Replies to the message with the reply string.
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. If it is decided to implement the rule, some customization is required.
Customization	The rule assumes that test CICS environments are named CICST*. This value must be adapted (in the rule ON statement) to match site conventions.

Table 84 CICS Journal Full—Production Environment Rule Structure (part 1 of 3)

Item	Description
Title	**CICS Journal Full—Production Environment**
Name	**DFH*4583**
Table	**CICS**
Message	DFH4583/DFHJC4583 aplid CICS {SYSTEM LOG | JOURNAL *nn*} {PRIMARY | SECONDARY | EMERGENCY} DATASET (DDNAME=DFHJ*nnx*) READY TO BE COPIED. REPLY *'Ynnx'* WHEN COPIED. and the job that issued the message is CICSP*.
Message Description	The CICS journal dataset indicated in the message has filled up and the journal was specified with the PAUSE option. The message waits for a response indicating that the journal dataset can be opened for output again. A correct reply must be received before CICS attempts to access the dataset for output again. Otherwise, all tasks using the specified journal are delayed until the reply is received.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%REPLY_STRING_*cicsname* The reply string indicated in the message. This string is used in the job JCL to identify the dataset to be copied. *cicsname* is the name of the CICS address space. %%REPLY_NO_*cicsname* Reply number of message DFH4583 / DFHJC4583. *cicsname* is the name of the CICS address space.

Table 84 CICS Journal Full—Production Environment Rule Structure (part 2 of 3)

Item	Description
Rule Logic	The rule is triggered when a CICS journal dataset becomes full and that journal is specified with the PAUSE option. The message waits for a system operator reply indicating that the dataset can be used again by CICS. The reply string indicated in the message contains the journal identifier and type. A Global AutoEdit variable is set to contain this string and is referenced by the job JCL. The CONTROL-O $GLOBAL member is included in the job JCL by %%LIBSYM and %%MEMSYM AutoEdit control statements. The rule sets a condition to trigger a pre-scheduled job in CONTROL-M. The job copies the applicable CICS journal datasets. The job must be defined as a cyclic job or cyclic started task (STC). It is triggered by adding the prerequisite condition or date CTO-*cicsname*-JRN-GO STAT. (For more details on the job scheduling definition, see Customization in this table.)
Rule Actions	■ Sets Global variable %%REPLY_STRING_*cicsname* to the reply string indicated in the message. ■ Sets Global variable %%REPLY_NO_*cicsname* to the message reply number. ■ Issues a command instructing CONTROL-O to write the Global variables. ■ Adds prerequisite condition or date CTO-cicsname-JRN-GO STAT. ■ Notifies user U-SYSCICS (the CICS administrator) that a job was sent to copy the CICS journal.
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.

Table 84 CICS Journal Full—Production Environment Rule Structure (part 3 of 3)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different automatic mechanism to clean CICS journals (for example, AUTOARCH option for CICS version 3.1 and later) is already implemented, the mechanism must be removed before testing this rule. By automating CICS journal dumping with CONTROL-O, the benefits of integrated operation are achieved. CONTROL-M tracks and controls the copy and clean job, and manages Quantitative resources used by the job. The rule assumes that production CICS environments are named CICSP*. This value must be adapted (in the rule ON statement) to match the site's conventions. Job scheduling definition and JCL for the CICS journal copy and clean job must be created. The SOLVSCHD and SOLVJCL Libraries contain a sample job scheduling definition and JCL to copy and clean CICS journal. These samples can be adapted to a site's conventions and requirements. Separate job scheduling definitions must be created for each production environment. The job must be defined as a cyclic job or cyclic started task (STC) with a MAXWAIT value of 99. It then only needs to be ordered once, but must not be removed manually from the CONTROL-M Active Jobs file. The MAXWAIT value of 99 ensures that the job is never removed from the Active Jobs file by the CONTROL-M New Day procedure. The cyclic job is always ready for submission. It is triggered by adding prerequisite condition or date CTO-*cicsname*-JRN-GO 0101 (where *cicsname* is the name of the CICS address space). When an execution of the job is completed, this condition is deleted. This prevents cyclic reinvoking of the job and ensures that the job is only invoked again if the rule is triggered again. Upon successful completion of the job, the prerequisite condition or date CTO-cicsname-JRN-REPLY STAT (where *cicsname* is the name of the CICS address space) must be added by CONTROL-M. This triggers an event rule that replies to the original message, notifying CICS that the journal dataset is available.

Table 85 CICS Journal Copied—Production Environment Rule Structure

Item	Description
Title	**CICS Journal Copied—Production Environment**
Name	**CICSP1JC**
Table	**CICS**
Message	`CICSP1JC`
Message Description	This Event rule replies to CICS message DFH4583 / DFHJC4583 after the journal dataset has been successfully copied.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	IN CTO-*cicsname*-JRN-REPLY STAT
Global Variables	`%%REPLY_STRING_`*`cicsname`* String to be issued as a reply. *cicsname* is the name of the CICS address space. `%%REPLY_NO_`*`cicsname`* Reply number of message DFH4583 / DFHJC4583. *cicsname* is the name of the CICS address space.
Rule Logic	This event rule is triggered upon successful completion of the CICS journal copy job. The rule replies to CICS message DFH4583, indicating that the journal dataset can be used again by CICS. Reply %%REPL_STRING_*cicsname* is issued to reply number %%REPLY_NO_*cicsname*. These Global variables are set by rule DFH4583 (in this solution).
Rule Actions	■ Replies to reply number %%REPLY_NO_*cicsname* with string %%REPLY_STRING_*cicsname* ■ Deletes prerequisite condition or date CTO-*cicsname*-JRN-REPLY STAT
Activating the Rule	Once scheduled, the rule is triggered when its IN prerequisite condition is added. The rule then deletes that condition and can be invoked again subsequently.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Create a similar rule for every production CICS environment. To create a new event rule, change the references to the CICS address space name in the rule actions and IN condition.

Chapter 12

DB2

DB2 solutions are designed to establish a link between DB2 and CONTROL-M jobs. This link is established using IOA prerequisite conditions that reflect DB2's status (up or down) and Quantitative resources that indicate on which computer DB2 is running (for follow-me support). This connection enables CONTROL-M to schedule certain jobs only when DB2 is active. Optionally, in a multi-CPU data center, jobs requiring DB2 services (follow-me jobs) can be submitted by CONTROL-M to the same CPU (that is, MVS system) where DB2 is operated. DB2 solutions also “shout” messages notifying the database administrator (DBA) of problems in the DB2 environment.

NOTE

All messages (DO SHOUT actions) in DB2 rules are sent to an INCONTROL user named U-SYSDBA. A user with this name must be defined in the IOA Dynamic Destination table (CTMDEST). For more information, see the Dynamic Destination Table chapter of the *INCONTROL for z/OS Administrator Guide*.

In many cases, rule definitions make use of the inverse IN condition feature. This feature activates rules only if the specified IN conditions are not set. (For more information about using inverse IN conditions, see the section on SolveWare implementation considerations in Chapter 1, “Introduction.”

Solutions Provided

SolveWare subject DB2 contains the following solutions:

- DB2 Startup and Termination

  Intercepts all messages related to DB2 initialization or termination and updates the appropriate IOA prerequisite conditions.

- DB2 Follow-me Job Management

Enables CONTROL-M to submit DB2-related jobs to the same CPU (that is, MVS system) where DB2 is operating.

- DB2 Problem Alerts

  Intercepts important DB2 error messages and notifies the database administrator of the problems.

- DB2 Rule Thresholds

  Handles exceeded thresholds of other DB2 rules.

DB2 Startup and Termination

At sites operating both CONTROL-M and DB2, the status of DB2 (up or down) impacts the start of jobs that run under CONTROL-M. Therefore, for the production DB2 in the system, the site must maintain two prerequisite conditions indicating DB2 status – CTO-DB2-UP STAT and CTO-DB2-DOWN STAT.

These two conditions are updated by CONTROL-O according to DB2 initialization and termination messages and are used as prerequisite conditions for jobs run under CONTROL-M that require DB2 status information.

NOTE

IOA DB2 prerequisite conditions and quantitative resources must also be updated at time of IPL to indicate that the various DB2 subsystems are down. For more information, see the section on initializing SolveWare in Chapter 2, “STARTSYS.”

Rules

The DB2 Startup and Termination solution includes the following rules:

- Notify CONTROL-M That DB2 Is Up
- Notify CONTROL-M That DB2 Is Down

Rules Structure

The following tables describe the structures of the DB2 Startup and Termination solution rules.

Table 86 Notify CONTROL-M That DB2 Is Up Rule Structure (part 1 of 2)

Item	Description
Title	**Notify CONTROL-M That DB2 Is Up**
Name	**DSN9022I**
Table	**DB2**
Message	`DSN9022I DSNYASCP `verb' NORMAL COMPLETION` where *verb* is – START DB2.
Message Description	Synchronous processing for command – START DB2 ended without error. DB2 is up and ready. "–" is a one-character identifier that identifies the DB2 subsystem.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	This rule, triggered by the above message, sets the IOA conditions that indicate DB2's status (up or down). This way, jobs in CONTROL-M that are waiting for DB2 to be up can be submitted and jobs that require DB2 to be down are not submitted.
Rule Actions	■ Deletes condition or date CTO-DB2-DOWN STAT ■ Adds condition or date CTO-DB2-UP STAT ■ Deletes condition or date CTO-DB2-DOWN-*smfid* STAT ■ Adds condition or date CTO-DB2-UP-*smfid* STAT
	Note: *smfid* is the SMF ID of the CPU.
Activating the Rule	Once scheduled, the rule is triggered whenever the above message is issued by the DB2 subsystem.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.

Table 86 Notify CONTROL-M That DB2 Is Up Rule Structure (part 2 of 2)

Item	Description
Customization	Each DB2 subsystem in the installation has a 1-character identifier and all commands that refer to this DB2 subsystem must start with this character. Adapt "–" in command –START DB2 (inside the rule definition) to identify the relevant (production) DB2 subsystem in your installation. If a different mechanism to control DB2 status (utility IOACND, CONTROL-M Event Manager, and so on) is already implemented, the mechanism must be removed before implementing this rule. Change the condition names either in this rule or in the existing job scheduling definitions, if both of the following are true: ■ The IOA environment already uses prerequisite conditions indicating that DB2 is up or down. For example, if such conditions are already referenced by CONTROL-M job scheduling definitions. ■ The prerequisite conditions have different names than the ones indicated in this rule.
	Note: As noted earlier, IOA DB2 prerequisite conditions must also be updated at time of IPL to indicate that the various DB2 subsystems are down. For details, see the section about initializing SolveWare in Chapter 2, "STARTSYS."

Table 87 Notify CONTROL-M That DB2 Is Down Rule Structure (part 1 of 2)

Item	Description
Title	**Notify CONTROL-M That DB2 Is Down**
Name	**DSN3100I**
Table	**DB2**

Message	Any of the following messages:
	
	`DSN3100I csect SUBSYSTEM ssss READY FOR START COMMAND`
	
	`DSN3104I csect - TERMINATION COMPLETE`
	
	`DSN3106I csect SUBSYSTEM STOPPED. COMMAND cmd IGNORED`

Message Description	DB2 subsystem is down. It can be started by issuing a start command.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Table 87 Notify CONTROL-M That DB2 Is Down Rule Structure (part 2 of 2)

Item	Description
Global Variables	None.
Rule Logic	This rule, triggered by any of the above messages, sets the IOA conditions that indicate DB2's status (up or down). This way, jobs in CONTROL-M that are waiting for DB2 to be down can be submitted and jobs that require DB2 to be up are not submitted.
Rule Actions	■ Deletes condition or date CTO-DB2-UP STAT ■ Adds condition or date CTO-DB2-DOWN STAT ■ Deletes condition or date CTO-DB2-UP-*smfid* STAT ■ Adds condition or date CTO-DB2-DOWN-*smfid* STAT
	Note: *smfid* represents the SMF ID of the CPU.
Activating the Rule	Once scheduled, the rule is triggered whenever one of the above messages is issued by the DB2 subsystem.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism to control DB2 status (utility IOACND, CONTROL-M Event Manager, and so on) is already implemented, the mechanism must be removed before implementing this rule. Change the condition names either in this rule or in the existing job scheduling definitions, if both of the following are true: ■ The IOA environment already uses prerequisite conditions indicating that DB2 is up or down. For example, if such conditions are already referenced by CONTROL-M job scheduling definitions. ■ The prerequisite conditions have different names than the ones indicated in this rule.
	Note: As noted earlier, IOA DB2 prerequisite conditions must also be updated at time of IPL to indicate that the various DB2 subsystems are down. For details, see the section about initializing SolveWare in Chapter 2, “STARTSYS.”

DB2 Follow-me Job Management

In a multi-CPU configuration operating both CONTROL-M and DB2, certain jobs must run on the same CPU on which DB2 is operating because they require DB2 services. These jobs are referred to as follow-me jobs.

For each CPU on which DB2 may operate, Quantitative resource CTO-DB2-CPU-ID is maintained by rules in this solution (ID is a 1-character identifier of the CPU). A quantity of 9999 is assigned to the resource corresponding to the CPU on which DB2 is operating; a quantity of 0000 is assigned to the corresponding resources on all other CPUs.

CONTROL-M job scheduling definitions for “follow-me” jobs must contain Quantitative resource CTO-DB2-CPU-$. The $ character is a mask. CONTROL-M replaces the $ character with the identifier of the CPU whose resource CTO-DB2-CPU-ID has a quantity of 9999. This CPU identifier is also assigned to the CONTROL-M AutoEdit system variable %%$SIGN, which is used in the JCL of the follow-me jobs to establish the class on which the job is submitted (on the same CPU where DB2 is operated).

For more information on Quantitative resources containing a $ sign, see the Job Production Parameters chapter in the *CONTROL-M for z/OS User Guide*. The technique of controlling the target computer by AutoEdit system variable %%$SIGN is described in the JCL and AutoEdit Facility chapter in the *CONTROL-M for z/OS User Guide*.

Rules

The DB2 Follow-me Job Management solution includes the following rules:

- Set the Follow-me Quantitative Resource
- Reset the Follow-me Quantitative Resource

Rules Structure

The following tables describe the structures of the DB2 Follow-me Job Management solution rules.

Table 88 Set the Follow-me Quantitative Resource Rule Structure (part 1 of 2)

Item	Description
Title	**Set the Follow-me Quantitative Resource**
Name	**DB2UP**
Table	**DB2**
Event	DB2UP
Event Description	This event rule sets the quantity of resource CTO-DB2-CPU-*id* (used by follow-me jobs) to 9999.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	%%CPUSEQ_*smfid* For each computer's SMF ID, a one-character identifier of the corresponding system. For example: %%CPUSEQ_SYS1 = 1 %%CPUSEQ_SYS2 = 2
Rule Logic	This rule is triggered when DB2 starts operating. The one-character identifier of the CPU is obtained from the SMF ID and the quantity of resource CTO-DB2-CPU-id for the current CPU is set to 9999.
Rule Actions	Sets Quantitative resource CTO-DB2-CPU-id to 9999.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.

Table 88 Set the Follow-me Quantitative Resource Rule Structure (part 2 of 2)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	For each CPU on which CONTROL-O is active, define a Global variable in the Global Variable list. Each variable must be defined in the format %%CPUSEQ_*smfid=c*, where *smfid* is the SMF ID of the CPU and *c* is a unique, 1-character identifier of the CPU. These variables can be defined manually by adding the variables to the CONTROL-O $GLOBAL member and issuing operator command F CONTROLO, LOADGLOBAL, or they can be defined by adding corresponding DO SET statements to a rule in the INITSLV rule table (see Chapter 2, "STARTSYS"). CONTROL-M job scheduling definitions and JCLs for follow-me jobs must be modified to enable CONTROL-M to dynamically determine the job target computer. SOLVSCHD and SOLVJCL libraries contain, respectively, a sample job scheduling definition and a sample JCL that serve as examples for defining such jobs. These samples can be adapted to site conventions and requirements.

Table 89 Reset the Follow-me Quantitative Resource (part 1 of 2)Rule Structure

Item	Description
Title	**Reset the Follow-me Quantitative Resource**
Name	**DB2DOWN**
Table	**DB2**
Event	DB2DOWN
Event Description	This event rule resets the quantity of resource CTO-DB2-CPU-*id* (used by follow-me jobs) to 0000.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations

Global Variables	%%CPUSEQ_*smfid*
	For each computer's SMF ID, a one-character identifier of the corresponding system.
	
	For example:
	
	`%%CPUSEQ_SYS1 = 1`
	`%%CPUSEQ_SYS2 = 2`

Table 89 Reset the Follow-me Quantitative Resource (part 2 of 2)Rule Structure

Item	Description
Rule Logic	This rule is triggered when DB2 stops operating. The one-character identifier of the CPU is obtained from the SMF ID and the quantity of resource CTO-DB2-CPU-id for the current CPU is reset to 0000.
Rule Actions	Reset Quantitative resource CTO-DB2-CPU-*id* to 0000.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	For each CPU on which CONTROL-O is active, define a Global variable in the Global Variable list. Each variable must be defined in the format %%CPUSEQ_smfid=c, where *smfid* is the SMF ID of the CPU and c is a unique, 1-character identifier of the CPU. These variables can be defined by manually by adding the variables to the CONTROL-O $GLOBAL member and issuing operator command F CONTROLO,LOADGLOBAL, or by adding corresponding DO SET statements to a rule in the INITSLV rule table (see Chapter 2, “STARTSYS”). CONTROL-M job scheduling definitions and JCLs for follow-me jobs must be modified to enable CONTROL-M to dynamically determine the job target computer. SOLVSCHD and SOLVJCL libraries contain, respectively, a sample job scheduling definition and a sample JCL that serve as examples for defining such jobs. These samples can be adapted to site conventions and requirements.

DB2 Problem Alert

The rules in this solution notify the database administrator of important problems in the DB2 environment.

Rules

The DB2 Problem Alert solution includes the following rules:

- Table Space Problem
- Threads Are Unavailable

Rules Structure

The following tables describe the structures of the DB2 Problem Alert solution rules.

Table 90 Table Space Problem Rule Structure

Item	Description
Title	**Table Space Problem**
Name	**DSNU086I**
Table	**DB2**
Message	DSNU086I *csect ttttt* I/O ERRORS ON SPACE=*dddddddd*. DATASET NUMBER=*nnn*. I/O ERROR PAGE RANGE=*aaaaaa,bbbbbb*
Message Description	One or more I/O errors occurred while accessing a table space.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	This rule is triggered by the above DB2 message. The rule “shouts” a message identifying the problematic table space name to the database administrator.
Rule Actions	■ Extracts the problematic table space name from the DB2 message. ■ Extracts the dataset number from the DB2 message. ■ “Shouts” a message notifying U-SYSDBA that the named table space is problematic (and probably full).
Activating the Rule	Once scheduled, the rule is triggered whenever the above message is issued by the DB2 subsystem.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1—little or no customization is required before implementation.

Table 91 Threads Are Unavailable Rule Structure

Item	Description
Title	**Threads Are Unavailable**
Name	**DSNC011I**
Table	**DB2**
Message	DSNC011I *xxxx* TRANSACTION ABENDED BECAUSE THREADS ARE UNAVAILABLE

Table 91 Threads Are Unavailable Rule Structure

Item	Description
Message Description	The transaction identified by *xxxx* abended because a thread connection to DB2 was not available for the transaction. This indicates that the maximum number of thread connections had already been reached before the current transaction.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	This rule is triggered when the above DB2 message is issued. The rule “shouts” the DB2 message text to the database administrator.
Rule Actions	“Shouts” message text notifying U-SYSDBA of the thread-unavailability problem.
Activating the Rule	Once ordered, the rule remains active until message DSNC011I exceeds a predefined threshold. (For more information regarding threshold handling, see the following section.)
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1—little or no customization is required before implementation.

DB2 Rule Thresholds

This solution handles message overload – situations in which a message appears on the console more times than is acceptable. If a DB2 message appears too often on the system console, threshold rules deactivate the relevant DB2 rule until the source of the problem is found and the problem corrected.

For more information regarding threshold handling, see “SolveWare Implementation Considerations” on page 24.

Rules

The DB2 Rule Thresholds solution includes the following rules:

- Handling Exceeded DB2 Thresholds
- Resetting DB2 Rule Threshold Conditions

Rules Structure

The following tables describe the structures of the DB2 Rule Thresholds solution rules.

Table 92 Handling Exceeded DB2 Thresholds Rule Structure (part 1 of 2)

Item	Description
Title	**Handling Exceeded DB2 Thresholds**
Name	**DSNC011I**
Table	**DB2**
Message	Either of the following messages: DSNC011I *xxxx* TRANSACTION ABENDED BECAUSE THREADS ARE UNAVAILABLE DSNU086I *csect ttttt* I/O ERRORS ON SPACE=*dddddddd*. DATASET NUMBER=*nnn*. I/O ERROR PAGE RANGE=*aaaaaa,bbbbbb*
Message Description	Messages handled by other rules comprising the DB2 solutions.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	To avoid message overload situations, this rule deactivates DB2 rules whose messages have exceeded a predetermined number of appearances in a period of time. These threshold values are defined for every message included in this rule in the threshold parameter APPEARED ### TIMES IN #### MINUTES. (The default is 5 times in 720 minutes.) To synchronize threshold handling correctly, this rule is assigned a higher PRIORITY value than the message rules that it monitors, and has a CONTINUE SEARCH value of Y (Yes). Deactivation of a message rule is achieved by adding the inverse IN prerequisite condition that is defined for the rule. To reactivate the deactivated message rule, the threshold conditions must be deleted. This can be done either manually or automatically by CONTROL-O (see the following section). Threshold conditions must be specified in the IGNORE list of the CONTROL-M CONTDAY procedure (see "SolveWare Implementation Considerations" on page 24).

Table 92 Handling Exceeded DB2 Thresholds Rule Structure (part 2 of 2)

Item	Description
Rule Actions	■ Notifies user U-SYSDBA that the message that has exceeded its threshold is no longer handled by CONTROL-O. ■ Sets the appropriate IN condition to deactivate the rule. The format of this condition is: ■ CTO-*msgid*-THRESH where *msgid* is the message ID of the specific message.
Activating the Rule	Once scheduled, the rule is triggered whenever one of the above messages exceeds its threshold.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Review each DB2 message monitored by this rule and determine appropriate threshold values. For each message included in this rule, adapt to site requirements the APPEARED ### TIMES IN #### MINUTES value, which specifies a number of appearances in a time period.

Table 93 Resetting DB2 Rule Threshold Conditions (part 1 of 2)Rule Structure

Item	Description
Title	**Resetting DB2 Rule Threshold Conditions**
Name	**RESDB2**
Table	**DB2**
Event	RESDB2
Event Description	This Event rule deletes all threshold conditions for all DB2 rules.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations
Global Variables	None.
Rule Logic	Using the INTERVAL parameter, this rule periodically deletes the threshold conditions of all DB2 rules in order to reactivate DB2 rules that exceeded their thresholds. For more information regarding the resetting of threshold conditions, see Customization in this table.
Rule Actions	■ Deletes condition or date CTO-DSNC011I-THRESH STAT ■ Deletes condition or date CTO-DSNU086I-THRESH STAT
Activating the Rule	Once scheduled, the rule is triggered periodically according to the INTERVAL parameter specification.

Table 93 Resetting DB2 Rule Threshold Conditions (part 2 of 2)Rule Structure

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is —some customization is required before implementation.
Customization	Adapt the INTERVAL parameter to site requirements: If you use the INTERVAL parameter without the TIME FROM parameter, the threshold conditions are deleted when the rule is ordered.
	Note: Threshold conditions must also be deleted at time of IPL (see "SolveWare Initialization" on page 31). Threshold conditions must be specified in the IGNORE list of the CONTROL-M CONTDAY procedure (see "SolveWare Implementation Considerations" on page 24).

Chapter 13

IMS

Solutions in the SolveWare subject, IMS, are designed to automate several management aspects of IMS (IBM's Information Management System).

Solutions Provided

SolveWare subject, IMS, contains the following solutions:

- IMS Startup and Termination

  Monitors the IMS startup and termination processes, intercepts relevant messages and updates the appropriate IOA prerequisite conditions.

- IMS Reply Automation

  Provides a general interface for issuing IMS commands. The current IMS reply number is maintained by CONTROL-O, enabling other tasks (such as CONTROL-M) in the system to easily issue reply commands to IMS.

IMS Startup and Termination

This solution handles IMS startup and termination.

A link between IMS and INCONTROL products is established by defining IOA prerequisite conditions that reflect IMS's status (up or down). Specifying these conditions as prerequisite IN conditions in IOA definitions makes processing dependent upon IMS being up or down.

NOTE

While startup or termination is in process—but not yet completed—the IOA environment considers IMS as both “not up” and “not down.” Therefore, IMS not being up is not necessarily the same as a IMS down status and vice versa.

IOA IMS prerequisite conditions must also be updated at time of IPL to indicate that the production IMS subsystem is down (see “SolveWare Initialization” on page 31.)

Rules

The IMS Startup and Termination solution includes the following rules:

- IMS Initialization Started
- IMS Initialization Completed
- IMS Termination Started
- IMS Termination Completed
- IMS Severe Termination (JES2 Only)
- Suppress S IMS if IMS Is Active

Rules Structure

The following tables describe the structures of the IMS Startup and Termination solution rules.

Table 94 IMS Initialization Started Rule Structure (part 1 of 2)

Item	Description
Title	**IMS Initialization Started**
Name	**IEF403I**
Table	**IMS**
Message	`IEF403I jjj STARTED { - TIME=hh.mm.ss}` when the message is issued from job IMS* (* is the generic mask character).
Message Description	Job *jjj* starts.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.

Table 94 IMS Initialization Started Rule Structure (part 2 of 2)

Item	Description
Rule Logic	As soon as IMS is started, the IOA environment is informed that IMS is no longer down.
Rule Actions	Deletes condition or date CTO-*imsname*-DOWN STAT (where *imsname* is the name of the IMS job or started task).
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls IMS conditions (utility IOACND, CONTROL-M Event Manager, and so on) is already implemented, the existing mechanism must be removed before implementing this rule. Change the name of the prerequisite condition deleted by this rule to match your previous definition, if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that IMS is down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule. If required, change the IMS job or started task name mask in field JNAME of the rule's ON block to conform to local naming conventions. The default-defined value is IMS*.

Table 95 IMS Initialization Completed Rule Structure (part 1 of 2)

Item	Description
Title	**IMS Initialization Completed**
Name	**DFS994I**
Table	**IMS**

Message	One of the following messages:
	
	`DFS994I rtype START COMPLETED`
	`DFS994I rtype EMERGENCY START COMPLETED`
	`DFS994I TAKEOVER COMPLETED`

Message Description	IMS has completed initialization and is up.
Basic Scheduling Parameters	Always schedule this rule.

Table 95 IMS Initialization Completed Rule Structure (part 2 of 2)

Item	Description
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	When IMS is completely up, the IOA environment is informed that IMS is up.
Rule Actions	■ Checks that the third word of the message is START (to distinguish between different meanings of message DFS994I). ■ Adds condition or date CTO-*imsname*-UP STAT where *imsname* is the name of the IMS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls IMS conditions (utility IOACND, CONTROL-M Event Manager, and so on) is already implemented, the existing mechanism must be removed before implementing this rule. Change the name of the prerequisite condition added by this rule to match your previous definition, if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that IMS is up. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule.

Table 96 IMS Termination Started Rule Structure (part 1 of 2)

Item	Description
Title	**IMS Termination Started**
Name	**DFS32571**
Table	**IMS**
Message	`DFS3257I ONLINE LOG CLOSED ON` *ddname* when the message contains string CLOSED in columns 21–26
Message Description	IMS shutdown has started and the Online Log Dataset (OLDS) is closed.

Table 96 IMS Termination Started Rule Structure (part 2 of 2)

Item	Description
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	When IMS termination has started, the IOA environment is informed that IMS is no longer up.
Rule Actions	Deletes condition or date CTO-*imsname*-UP STAT where *imsname* is the name of the IMS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls IMS conditions (utility IOACND, CONTROL-M Event Manager, and so on) is already implemented, the existing mechanism must be removed before implementing this rule. Change the name of the prerequisite condition deleted by this rule to match your previous definition, if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that IMS is up. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule.

Table 97 IMS Termination Completed Rule Structure (part 1 of 3)

Item	Description
Title	**IMS Termination Completed**
Name	**IEF404I**
Table	**IMS**

Table 97 IMS Termination Completed Rule Structure (part 2 of 3)

Item	Description

Message	Either of the following messages:
	
	`IEF404I jjj ENDED{ - TIME=hh.mm.ss}`
	
	when the message is issued from job IMS* (* is the generic mask character).
	
	`DFS994I IMS SHUTDOWN COMPLETEDt`
	
	when the message contains string SHUTDOWN COMPLETED in columns 12–27.

Message Description	■ IEF404I – Job *jjj* ended ■ DFS994I – IMS has completed termination.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	When IMS termination is completed, the IOA environment is informed that IMS is down.
Rule Actions	■ Deletes condition or date CTO-*imsname*-UP STAT where *imsname* is the name of the IMS job or started task. ■ Adds condition or date CTO-*imsname*-DOWN STAT where *imsname* is the name of the IMS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.

Table 97 IMS Termination Completed Rule Structure (part 3 of 3)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls IMS conditions (utility IOACND, CONTROL-M Event Manager, and so on) is already implemented, the existing mechanism must be removed before implementing this rule. Change the name of the prerequisite condition added by this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that IMS is down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule. If required, change the IMS job or started task name mask in field JNAME of the rule's ON block to conform to local naming conventions. The default-defined value is IMS*.

Table 98 IMS Severe Termination Rule Structure (part 1 of 2)

Item	Description
Title	**IMS Severe Termination**
Name	**$HASP310**
Table	**IMS**
Message	`$HASP310 jj TERMINATED AT END OF MEMORY` when the message contains string IMS in columns 10 – 12.
Message Description	Abend of job *jjj* was so severe that no Recovery Termination Manager (RTM) processing occurred.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	For severe IMS abends, the IOA environment is informed that IMS is no longer up and that IMS is down.

Table 98 IMS Severe Termination Rule Structure (part 2 of 2)

Item	Description
Rule Actions	■ Deletes condition or date CTO-*imsname*-UP STAT where *imsname* is the name of the IMS job or started task that abended. ■ Adds condition or date CTO-*imsname*-DOWN STAT where *imsname* is the name of the IMS job or started task that abended.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls IMS conditions (utility IOACND, CONTROL-M Event Manager, and so on) is already implemented, the existing mechanism must be removed before implementing this rule. Change the names of the prerequisite conditions added or deleted by this rule to match your existing definitions if both of the following are true: ■ The IOA environment already contains prerequisite conditions indicating that IMS is up or down. For example, if such conditions are already referenced by CONTROL-M job scheduling definitions. ■ The prerequisite conditions have different names than the ones indicated in this rule.
Note: This rule applies only at JES2 sites.	

Table 99 Suppress S IMS if IMS Is Active Rule Structure (part 1 of 2)

Item	Description
Title	**Suppress S IMS if IMS Is Active**
Name	**S IMS**
Table	**IMS**
Command	S IMS
Command Description	This command starts IMS (the rule suppresses the command).
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	IN !CTO-*imsname*-DOWN STAT

Table 99 Suppress S IMS if IMS Is Active Rule Structure (part 2 of 2)

Item	Description
Global Variables	None.
Rule Logic	If command S IMS is issued when IMS is already up, the new started task is immediately shut down and normal termination messages are issued. These messages, however, give the appearance that IMS has gone down and therefore may incorrectly trigger rules. This rule prevents this situation by suppressing the S IMS command if IMS is already up. In this way, issuance of the termination messages is prevented and rules are not incorrectly triggered. This rule serves as a sample rule. A similar rule must be created for each IMS environment in the site.
Rule Actions	■ Suppresses the command. ■ Sends a message to user U-SHFTOPER indicating that command S IMS was suppressed.
Activating the Rule	Once scheduled, the rule becomes active when prerequisite condition or date CTO-*imsname*-DOWN STAT is deleted (where *imsname* is the name of the IMS address space).
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Change the name of the prerequisite condition indicated in this rule to match your previous definition, if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that IMS is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule. The rule must be duplicated for each IMS in the site. To adapt each rule, the IMS name in the rule's ON statement and IN condition must be changed.

IMS Reply Automation

This solution facilitates issuing IMS reply commands. The current IMS reply number is maintained in a CONTROL-O Global variable. A new operator command uses this Global variable for issuing replies. This enables operators, online users and batch jobs to issue replies without having to determine the current IMS reply number.

Rules

The IMS Reply Automation solution includes the following rules:

- Keep Last IMS Reply Number
- Issue an IMS Reply Command

Rules Structure

The following tables describe the structures of the IMS Reply Automation solution rules.

Table 100 Keep Last IMS Reply Number Rule Structure (part 1 of 2)

Item	Description
Title	**Keep Last IMS Reply Number**
Name	**DFS996I**
Table	**IMS**
Message	`DFS996I IMS READY`
Message Description	IMS startup is completed. This message is issued as a WTOR (Write to Operator with Reply) message, thereby enabling entry of data through the system console.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%IMS_REPLY_*imsname* IMS open reply number.
Rule Logic	As soon as the above message appears on the console, the message reply number is updated in a CONTROL-O Global variable. This variable is later accessed by another CONTROL-O rule that issues reply commands to IMS.
Rule Actions	Assigns the message reply number (system variable %%REPLY) to Global variable %%IMS_REPLY_*imsname*.

Table 100 Keep Last IMS Reply Number Rule Structure (part 2 of 2)

Item	Description
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is 1—little or no customization is required.

Table 101 Issue an IMS Reply Command Rule Structure (part 1 of 2) (part 1 of 2)

Item	Description
Title	**Issue an IMS Reply Command**
Name	**IMSCMD-***
Table	**IMS**
Command	`IMSCMD-`*`imsname,imsdata`* where *imsname* is the name of the IMS job or started task name and *imsdata* is the data to be entered.
Command Description	This command rule is used to issue a command through the IMS open reply message (WTOR).
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%IMS_REPLY_*imsname* IMS *open reply number*.
Rule Logic	This command uses the IMS reply number, which is stored in a Global variable (for each IMS subsystem), to issue an IMS command. The command is especially useful when an IMS command is to be issued from a source other than the console itself. For example, batch jobs or started tasks can use this command to issue IMS commands without having to determine the currently open IMS reply number. In this way, IMS commands can be defined as part of the automatic production flow under CONTROL-M.
Rule Actions	■ Parses command text to extract the IMS job name and the data to be entered. ■ Using the IMS reply number (Global variable %%IMS_REPLY_%%*imsname*), issues the appropriate reply command.

Table 101 Issue an IMS Reply Command Rule Structure (part 2 of 2) (part 2 of 2)

Item	Description
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for this rule. The SolveWare category for this rule is 1—little or no customization is required.

Chapter 14

OMEGAMON

Solutions in SolveWare subject OMEGAMON are designed to help utilize the problem detection and resolution capabilities of the OMEGAMON performance monitor.

These solutions consist of CONTROL-O rules and CONTROL-O KeyStroke OpenAccess (KOA) scripts. KOA scripts can interface directly with VTAM applications. They are invoked either by rules or by other KOA scripts. For more information on KOA scripts, see the KeyStroke Language chapter in the *CONTROL-O User Guide*.

NOTE

The KOA scripts provided with these solutions can serve as models for writing other KOA scripts to be used with OMEGAMON.

Some messages (DO SHOUT actions) in the OMEGAMON rules are sent to INCONTROL user U-SYSADMIN. This user must be defined in the IOA Dynamic Destination table (CTMDEST). For more information, see the Dynamic Destination Table chapter in the *INCONTROL for z/OS Administrator Guide*.

Solutions Provided

SolveWare subject OMEGAMON contains the following solutions:

- OMEGAMON Exceptions

  Notifies the responsible user of specific important exceptions detected by OMEGAMON.

- Reserved Dataset Handling

  Locates the users, jobs or started tasks holding datasets that are required by a production job in Exclusive mode and sends an appropriate message to the user and/or to the production manager.

OMEGAMON Exceptions

This solution handles various OMEGAMON exceptions displayed in response to OMEGAMON command LEXSY.

A rule is used to start a KOA script that logs onto OMEGAMON and issues OMEGAMON command LEXSY with automatic refresh. This KOA script remains logged onto OMEGAMON until the next IPL or until the user manually terminates the KOA script.

The KOA script periodically analyzes the exceptions indicated by OMEGAMON in the refreshed screen for new exceptions of interest. For each specific exception of interest, a message is sent to the responsible user.

This solution contains one rule to start the KOA script and one rule to terminate it. The solution also contains the KOA script.

NOTE

The prerequisite conditions for the rules included in this solution (CTO-KOAOM-GO and CTO-KOAOM-STOP) must also be reset at IPL. For more information, see the section on initializing SolveWare in Chapter 2, “STARTSYS.”

Rules

The OMEGAMON Exceptions solution includes the following rules:

- Start OMEGAMON Exceptions KOA Script
- Terminate OMEGAMON Exceptions KOA Script

Rules Structure

The following tables describe the structures of the OMEGAMON Exceptions solution rules, as well as the following KOA script:

- Log onto OMEGAMON and Analyze Exceptions KOA Script.

Table 102 Start OMEGAMON Exceptions KOA Script Rule Structure (part 1 of 2)

Item	Description
Title	**Start OMEGAMON Exceptions KOA Script**
Name	**STRTEXCP**
Table	**OMEGAMON**
Event	STRTEXCP
Event Description	This Event rule activates a KOA script that logs onto OMEGAMON, performs OMEGAMON command LEXSY and analyzes the results of the command. The rule is usually triggered only once, immediately following IPL, but the KOA script remains logged onto OMEGAMON and continues analyzing exceptions until the next IPL.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	IN CTO-KOAOM-GO STAT
Global Variables	%%CTO_KOAOM_STOP
Rule Logic	This rule is usually triggered automatically, immediately following IPL. The rule starts KOA script OMLEXSY, which logs onto OMEGAMON and issues OMEGAMON command LEXSY with automatic refresh. The KOA script remains logged onto OMEGAMON until the next IPL, or until the user manually terminates the script (using rule STOPEXCP in this solution). The rule deletes its own prerequisite IN condition CTO-KOAOM-GO STAT so that the rule can be re-invoked (by adding this condition again).
Rule Actions	■ Sets CONTROL-O Global variable %%CTO-KOAOM-STOP to NO. ■ Activates KOA *script* OMLEXSY. ■ Deletes condition CTO-KOAOM-GO STAT.
Activating the Rule	The rule is triggered when its prerequisite IN condition CTO-KOAOM-GO STAT is set. For details about how to stop KOA script OMLEXSY, see the following section.

Table 102 Start OMEGAMON Exceptions KOA Script Rule Structure (part 2 of 2)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization of the KOA script is required before implementation.
Customization	The rule itself requires little or no customization, but KOA script OMLEXSY must be carefully examined and adapted to site requirements.

Table 103 Terminate OMEGAMON Exceptions KOA Script Rule Structure

Item	Description
Title	**Terminate OMEGAMON Exceptions KOA Script**
Name	**STOPEXCP**
Table	**OMEGAMON**
Event	STOPEXCP
Event Description	This Event rule terminates the KOA script that detects OMEGAMON exceptions, by setting Global CONTROL-O variable %%CTO_KOAOM_STOP to YES.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	IN CTO-KOAOM-STOP STAT
Global Variables	%%CTO_KOAOM_STOP
Rule Logic	This rule must be triggered in order to terminate KOA script OMLEXSY. The rule sets CONTROL-O Global variable %%CTO_KOAOM_STOP to YES, which indicates that a user wants to terminate the KOA script. KOA script OMLEXSY periodically checks if Global variable %%CTO_KOAOM_STOP is set to YES; if it is, the script immediately logs off of OMEGAMON. The rule deletes its own prerequisite IN condition CTO-KOAOM-STOP STAT so that the rule can be re-invoked (by adding this condition again).
Rule Actions	■ Sets CONTROL-O Global variable %%CTO-KOAOM-STOP to YES. ■ Deletes its own prerequisite IN condition CTO-KOAOM-STOP STAT. ■ Issues a WRITEGLOBAL command to the CONTROL-O Monitor.

Table 103 Terminate OMEGAMON Exceptions KOA Script Rule Structure

Item	Description
Activating the Rule	The rule is triggered by manually setting its prerequisite IN condition CTO-KOAOM-STOP STAT.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1—little or no customization is required before implementation.

Table 104 Log onto OMEGAMON and Analyze Exceptions Script Structure (part 1 of 2)

Item	Description
Title	**Log onto OMEGAMON and Analyze Exceptions**
Name	**OMLEXSY**
KOA Script Description	This KOA script logs onto OMEGAMON, enters auto-refresh mode, and performs OMEGAMON command LEXSY (which displays exceptions detected by OMEGAMON). The script “shouts” a message to the responsible user when certain OMEGAMON exceptions are detected.
Activating the KOA Script	The KOA script is activated by rule STRTEXCP in this solution.
Parameters	None.
Global Variables	%%CTO_KOAOM_STOP
KOA Script Logic	The KOA script logs onto OMEGAMON and issues OMEGAMON command LEXSY with automatic refresh. In response, OMEGAMON displays a list of current system-wide exceptions, which is periodically refreshed. (For further details, see the *OMEGAMON/OS/390 User Guide*.) Exceptions considered of special interest are specified within the KOA script. The list of OMEGAMON exceptions is analyzed by the KOA script and when a specified exception is detected, the script “shouts” a message to the responsible user. The script periodically checks if Global variable %%CTO_KOAOM_STOP is set to YES; if it is, the script immediately logs off from OMEGAMON. As long as Global variable %%CTO_KOAOM_STOP is set to NO, the script periodically checks the refreshed exception list for new exceptions of interest.
	Note: Although an exception reappears in each refresh until the problem is corrected, KOA shouts the message for that exception only once – for that occurrence of the problem. (This is controlled through CONTROL-O variables %%CTO_%%EXCPID_FOUND and %%CTO_%%EXCPID_SENT.)

Table 104 Log onto OMEGAMON and Analyze Exceptions Script Structure (part 2 of 2)

Item	Description
Recommended Category	The SolveWare category for this KOA script is 3—the script is provided as an example. Some customization is needed if the script is to be implemented.
Customization	New exceptions of interest can be added to, or existing exceptions deleted from, the sample KOA scripts. SHOUT message destinations can be changed. Locate the SHOUT messages in the KOA scripts and determine, for each kind of exception, the TSO-user who is to receive this type of message. Command PAUSE specifies the number of seconds to wait between analyses of the (refreshed) list. The default value provided in the script is one minute. This value can be modified according to site requirements. You can use command MAXCOMMAND during the test period; however, remember to remove this limitation before implementing this solution to enable it to be active 24 hours a day.

Reserved Dataset Handling

This solution locates the users, jobs, or started tasks holding datasets that are required by a production job in Exclusive mode and sends appropriate message to the users and / or to the production manager.

Rules

The Reserved Dataset Handling solution includes the Respond to Reserved Dataset Message of a Production Job rule.

Rules Structure

The following tables describe the structures of the Reserved Dataset Handling solution rules, as well as the following KOA script:

- Log onto OMEGAMON and Perform LOC Command

Table 105 Respond to Reserved Dataset Message of a Production Rule Structure

Item	Description
Title	**Respond to Reserved Dataset Message of a Production**
Name	**IEF863I**
Table	**OMEGAMON**
Message	`IEF863I DSN=`*dsn* where JNAME begins with P.
Message Description	A job is waiting for one or more datasets that are not available. Message IEF863I is issued listing the unavailable datasets.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	The rule is triggered when a production job (that is, a job whose name starts with P) is waiting for unavailable datasets required in Exclusive mode. The rule calls KOA script OMLOC. The KOA script locates the users, jobs or started tasks holding the datasets and “shouts” appropriate messages to those users and / or to the production manager.
Rule Actions	Suppresses message IEF863I from the console. Calls KOA script OMLOC.
Activating the Rule	Once scheduled, the rule is triggered whenever message IEF863I is issued by any job whose name starts with a P (that is, a production job).
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Check if the production job names in your site begin with a P. If not, adjust the rule according to site naming conventions for production jobs. KOA script OMLOC may also require adaptation to site requirements.

Table 106 Log onto OMEGAMON and Perform the LOC Command Script Structure (part 1 of 2)

Item	Description
Title	**Log onto OMEGAMON and Perform the LOC Command**
Name	**OMLOC**

Table 106 Log onto OMEGAMON and Perform the LOC Command Script Structure (part 2 of 2)

Item	Description
KOA Script Description	This KOA script is invoked by rule IEF863I. The rule passes to the script a parameter that identifies a held dataset required in Exclusive mode by a production job. The script logs onto OMEGAMON and performs OMEGAMON command LOC, which identifies the users, jobs or started tasks holding the dataset. The script then “shouts” a message to TSO users holding the dataset and to the production manager for each job or started task that holds the dataset.
Activating the KOA Script	The KOA script is activated by rule IEF863I.
Parameters	■ %A1 Name of the required (held) dataset. ■ %A2 Name of the production job that produced message IEF863I.
Global Variables	None.
KOA Script Logic	This KOA script logs onto OMEGAMON and performs OMEGAMON command LOC. LOC locates all holders of the dataset identified in parameter %A1, which is passed to the KOA script by its initiating rule. The script checks the response from OMEGAMON. For each TSO user holding the dataset, a message is “shouted” to the user. For each job or started task holding the dataset, a message is “shouted” to the production manager.
Recommended Mode or Category	The SolveWare category for this KOA script is 3—the script is provided as an example. Implementation of the KOA script requires little or no customization.
Customization	Locate the SHOUT messages to the production manager in the KOA script and modify the destination to the TSO user ID of the production manager.

Chapter 15

TESTJOBS

The solutions in SolveWare subject TESTJOBS handle certain error situations arising from non-production jobs. Such a job is usually the responsibility of the job owner (that is, the user that submitted the job). The job owner is notified by CONTROL-O of job errors as they occur.

TESTJOBS solutions must not handle production jobs, since these are already monitored by CONTROL-M.

Solutions Provided

SolveWare subject TESTJOBS contains the following solutions:

- Test Job Failures—Option 1

  Notifies test job owner of problems concerning the job. The user ID of the job owner is derived from part of the job name.

- Test Job Failures—Option 2

  Notifies test job owner of problems concerning the job. The user ID of the job owner is determined by predefined Global variables.

Test Job Failures–Option 1

When certain problems occur with a test job, this solution notifies the owner. The rules in this solution assume that the user ID of the job owner is part of the job name.

If the user ID of the job owner cannot be obtained by a %%SUBSTR operation on the job name, see "Test Job Failures—Option 2" on page 209.

Rules

The Test Job Failures–Option 1 solution includes the following rules:

- Test Job Abended—Option 1
- Test Job Not Run—JCL Error—Option 1

Rules Structure

The following tables describe the structures of the Test Job Failures–Option 1 solution rules:

Table 107 Test Job Abended—Option 1 Rule Structure (part 1 of 2)

Item	Description
Title	**Test Job Abended – Option 1**
Name	**IEF450I**
Table	**TESTJOB1**
Message	`IEF450I jjj {ppp} sss - ABEND {Scde | Ucde} REASON=xxxxxxxx TIME=hh.mm.ss`
Message Description	Job *jjj* terminated abnormally.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.

Table 107 Test Job Abended—Option 1 Rule Structure (part 2 of 2)

Item	Description
Rule Logic	If a test job abends, this rule notifies the job owner immediately. The rule is triggered if the job that issued the message is named according to certain test job naming conventions. The rule assumes that the TSO user ID of job owner is the prefix of the job name. (For more details concerning rule assumptions, see Customization in this table.)
Rule Actions	■ Sets variable %%UNAME to the prefix of the job name. ■ Sends a message notifying TSO user %%UNAME of the abend.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Some customization is needed if the rule is to be implemented.
Customization	The rule makes several assumptions regarding test job naming conventions. These assumptions must be adapted to site conventions: ■ Test jobs are assumed to be prefixed T*. Adapt the JNAME field in the rule's ON statement to site conventions. The rule is triggered only for jobs matching the JNAME field specification. ■ The TSO user ID of the job owner is assumed to be the first four characters of the job name. Adapt the rule actions (setting variable %%UNAME) to site conventions.

Table 108 Test Job Not Run—JCL Error—Option 1 Rule Structure (part 1 of 2)

Item	Description
Title	**Test Job Abended—Option 1**
Name	**IEF4521**
Table	**TESTJOB1**

Message	Either of the following messages:
	
	`IEF452I jjj JOB NOT RUN - JCL ERROR`
	`IEF453I jjj JOB NOT RUN - JCL ERROR`

Message Description	Job *jjj* failed due to a JCL error.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.

Table 108 Test Job Not Run—JCL Error—Option 1 Rule Structure (part 2 of 2)

Item	Description
Rule Logic	If a test job fails as a result of a JCL syntax error, this rule notifies the job owner immediately. The rule is triggered if the job that issued the message is named according to certain test job naming conventions. The rule assumes that the TSO user ID of job owner is the prefix of the job name. (For more details concerning rule assumptions, see Customization in this table.)
Rule Actions	■ Sets variable %%UNAME to the prefix of the job name. ■ Sends a message notifying TSO user %%UNAME of the JCL error.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Some customization is needed if the rule is to be implemented.
Customization	The rule makes several assumptions regarding test job naming conventions. These assumptions must be adapted to site conventions: ■ Test jobs are assumed to be prefixed T*. Adapt the JNAME field in the rule's ON statements to site conventions. The rule is triggered only for jobs matching the JNAME field specifications. ■ The TSO user ID of the job owner is assumed to be the first four characters of the job name. Adapt the rule actions (setting variable %%UNAME) to site conventions.

Test Job Failures—Option 2

When problems occur with a test job, this solution notifies the owner. The rules in this solution determine the user ID of the job owner by predefined Global variables that match the first three or four characters of the job name.

Rules

The Test Job Failures–Option 2 solution includes the following rules:

- Test Job Abended—Option 2
- Test Job Not Run—JCL Error—Option 2
- Initialize Job Owner Global Variables

Rules Structure

The following tables describe the structures of the Test Job Failures–Option 2 solution rules:

Table 109 Test Job Abended—Option 2 Rule Structure (part 1 of 3)

Item	Description
Title	**Test Job Abended—Option 2**
Name	**IEF450I**
Table	**TESTJOB2**
Message	IEF450I *jjj* {*ppp*} *sss* - ABEND {Scde | Ucde} REASON=*xxxxxxxx* TIME=*hh.mm.ss*
Message Description	Job *jjj* terminated abnormally.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%JOB_OWNER_*pref* TSO user ID of the job owner, where *pref* is the 3-character to 4-character prefix derived from the job name.

Table 109 Test Job Abended—Option 2 Rule Structure (part 2 of 3)

Item	Description
Rule Logic	If a test job abends, this rule notifies the job owner immediately. The rule checks the first four characters of the job name against a list of Global variables containing TSO user ID's. The variable names are in the format %%JOB_OWNER_*pref*, where *pref* is the job name prefix. If a match is found, the value obtained from the matching variable is used as the DO SHOUT destination. If no match is found, the same check is performed for the first three characters of the job name. If a match is found, that matching variable is used as the DO SHOUT destination. If no match is found for both 3-character and 4-character prefixes, the DO SHOUT action is skipped. For more details concerning Global variables referenced by the rule, see Customization in this table.
Rule Actions	■ Sets variable %%PREF3 to the 3-character job name prefix. ■ Sets variable %%PREF4 to the 4-character job name prefix. ■ Sets variable %%OWNER to either %%JOB_OWNER_%%PREF4 or %%JOB_OWNER_PREF3 depending on which of these variables is defined. If neither is defined, %%OWNER is not set. ■ If %%OWNER is defined, sends a message notifying TSO user %%OWNER of the JCL error.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.

Table 109 Test Job Abended—Option 2 Rule Structure (part 3 of 3)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Some customization is needed if the rule is to be implemented.
Customization	The rule seeks the job owner user ID that matches the job prefix in the Global Variable list. These variables must be adapted to site requirements. This can be done either manually by adding variables to the CONTROL-O $GLOBAL member and issuing operator command F CONTROLO,LOADGLOBAL, or by implementing event rule JOBOWNER (in this solution). The format in the Global Variable list is %%JOB_OWNER_*pref* = user-ID where *pref* is the job name prefix. For example, if jobs prefixed with PRD are owned by TSO user PR01, then the Global Variable list entry must be %%JOB_OWNER_PRD = PR01

Table 110 Test Job Not Run—JCL Error—Option 2 Rule Structure (part 1 of 3)

Item	Description
Title	**Test Job Abended—Option 2**
Name	**IEF4521**
Table	**TESTJOB2**

Message	Either of the following messages:
	
	`IEF452I jjj JOB NOT RUN - JCL ERROR`
	`IEF453I jjj JOB NOT RUN - JCL ERROR`

Message Description	Job *jjj* failed due to a JCL error.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%JOB_OWNER_*pref* TSO user ID of the job owner, where *pref* is the 3-character to 4-character prefix derived from the job name.

Table 110 Test Job Not Run—JCL Error—Option 2 Rule Structure (part 2 of 3)

Item	Description
Rule Logic	If a test job fails as a result of a JCL syntax error, this rule notifies the job owner immediately. The rule checks the first four characters of the job name against a list of Global variables containing TSO user ID's. The variable names are in the format %%JOB_OWNER_*pref*, where *pref* is the job name prefix. If a match is found, the value obtained from the matching variable is used as the DO SHOUT destination. If no match is found, the same check is performed for the first three characters of the job name. If a match is found, that matching variable is used as the DO SHOUT destination. If no match is found for both 3-character and 4-character prefixes, the DO SHOUT action is skipped. For more details concerning Global variables referenced by the rule, see Customization in this table.
Rule Actions	■ Sets variable %%PREF3 to the 3-character job name prefix. ■ Sets variable %%PREF4 to the 4-character job name prefix. ■ Sets variable %%OWNER to either: %%JOB_OWNER_%%PREF4 or %%JOB_OWNER_PREF3 Depends on which of these variables is defined. If neither is defined, %%OWNER is not set. ■ If %%OWNER is defined, sends a message notifying TSO user %%OWNER of the JCL error.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.

Table 110 Test Job Not Run—JCL Error—Option 2 Rule Structure (part 3 of 3)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—the rule is provided as an example. Some customization is needed if the rule is to be implemented.
Customization	The rule seeks the job owner user ID that matches the job prefix in the Global Variable list. These variables must be adapted to site requirements. This can be done either manually by adding variables to the CONTROL-O $GLOBAL member and issuing operator command F CONTROLO,LOADGLOBAL, or by implementing event rule JOBOWNER (in this solution). The format in the Global Variable list is %%JOB_OWNER_*pref*=user-ID where *pref* is the job name prefix. For example, if jobs prefixed with PRD are owned by TSO user PR01, then the Global Variable list entry must be %%JOB_OWNER_PRD=PR01

Table 111 Initialize Job Owner Global Variables Rule Structure (part 1 of 2)

Item	Description
Title	**Initialize Job Owner Global Variables**
Name	**JOBOWNER**
Table	**TESTJOB2**
Event	JOBOWNER
Event Description	This Event rule sets the values of Global variables referenced by other rules in this solution.
Basic Scheduling Parameters	Schedule this rule if initialization or update of job-owner Global variables is desired.
Runtime Scheduling Parameters	PRIORITY 20
Global Variables	%%JOB_OWNER_*pref* TSO user ID of the job owner, where *pref* is the 3-character to 4-character prefix derived from the job name.

Table 111 Initialize Job Owner Global Variables Rule Structure (part 2 of 2)

Item	Description
Rule Logic	This rule sets the values of job-owner Global variables (%%JOB_OWNER_*pref*). These variables are used by the other rules in this solution to obtain the TSO user ID of a job with a certain prefix.
Rule Actions	■ Sets Global variable %%JOB_OWNER_SY01 to SY01 ■ Sets Global variable %%JOB_OWNER_IOA to SY01 ■ Sets Global variable %%JOB_OWNER_ADA to SY05
Activating the Rule	Once ordered, the rule is executed.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—this rule is provided as an example. Implementation of the rule requires some customization.
Customization	Adapt the rule to site requirements by changing the Global Variable list to reflect actual device definitions. A DO SET statement must be defined for every job name prefix to be identified by this solution. The format of the DO SET statement is `%%JOB_OWNER_`*`pref=user-ID`* where ■ *pref* is the job name prefix ■ *user-ID* is the TSO user ID of the job owner For example, if jobs prefixed with PRD are owned by TSO user PR01, the DO SET statement must be defined as `DO SET=%%JOB_OWNER_PRD=PR01` Changes made in this rule take effect when the rule is scheduled again. To remove previously defined Global variables from CONTROL-O, they must be removed both from rule definitions and from the CONTROL-O $GLOBAL member.

Chapter 16

ADABAS

ADABAS solutions are designed to keep track of the status and workload of ADABAS in order to enable integration with CONTROL-M and to automate the cleaning of ADABAS protection log. Integration achieved through IOA prerequisite conditions is useful for scheduling jobs that use ADABAS only when ADABAS is active.

NOTE

Some messages (DO SHOUT actions) in the ADABAS rules are sent to INCONTROL users (U-SYSDBA or U-SHIFTOPER). Users with these names can be defined in the IOA Dynamic Destination table (CTMDEST). For details, see the considerations chapter of the INCONTROL for z/OS Installation Guide.

Some rule definitions make use of the inverse IN condition feature. This feature activates rules only if the specified IN conditions are not set. For more information about using inverse IN conditions, see the section on SolveWare implementation considerations in Chapter 1, “Introduction.”

Solutions Provided

SolveWare subject ADABAS contains the following solutions:

- ADABAS Startup and Termination

  Intercepts all messages related to ADABAS initialization or termination and updates the appropriate IOA prerequisite conditions.

- Copy and Clean ADABAS Protection Log

  Handles situations where the ADABAS protection log dataset becomes full. A copy and clean job is submitted automatically to prevent loss of information.

ADABAS Startup and Termination

This solution handles ADABAS startup and termination.

At sites operating both CONTROL-M and ADABAS, the status of ADABAS (up or down) impacts the start of jobs that run under CONTROL-M. Therefore, such a site must maintain two prerequisite conditions indicating ADABAS status – CTO-ADABAS-UP STAT and CTO-ADABAS-DOWN STAT.

These two conditions are automatically updated by CONTROL-O according to ADABAS initialization and termination messages. They are used as prerequisite conditions for jobs run under CONTROL-M that require ADABAS status information.

NOTE

While startup or termination is in process—but not yet completed—the IOA environment considers ADABAS as both “not up” and “not down.” Therefore, ADABAS not being up is not necessarily the same as ADABAS being down, and vice versa.

Rules

The ADABAS Startup and Termination solution includes the following rules:

- ADABAS MPM*xx* Initialization Started
- ADABAS Initialization Completed
- ADABAS Termination Started
- ADABAS MPM*xx* Termination Completed
- Suppress S MPM*xx* if ADABAS Is Active

Rules Structure

The following tables describe the structures of the ADABAS Startup and Termination solution rules.

Table 112 ADABAS MPM*xx* Initialization Started Rule Structure (part 1 of 2)

Item	Description
Title	**ADABAS MPM*xx* Initialization Started**
Name	IEF403I
Table	ADABAS

Table 112 ADABAS MPM*xx* Initialization Started Rule Structure (part 2 of 2)

Item	Description
Message	`IEF403I jjj STARTED { - TIME=hh.mm.ss}` when the message is issued from job MPM*.
Message Description	Job jjj starts.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	As soon as ADABAS is started, the IOA environment is informed that ADABAS is no longer down.
Rule Actions	Deletes condition or date CTO-adabasname-DOWN STAT, where *adabasname* is the name of the ADABAS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls ADABAS conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that ADABAS is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule. The rule assumes that the name of the production ADABAS environments starts with MPM. This prefix may need to be changed (in the rule's ON statement) in order to match site conventions.

Table 113 ADABAS Initialization Completed Rule Structure

Item	Description
Title	**ADABAS Initialization Completed**
Name	ADAN01

Table 113 ADABAS Initialization Completed Rule Structure

Item	Description
Table	ADABAS
Message	`ADAN01 adabasname time ADABAS IS ACTIVE`
Message Description	ADABAS has completed initialization and is up.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	As soon as ADABAS initialization has been completed, the IOA environment is informed that ADABAS is up.
Rule Actions	Adds condition or date CTO-adabasname-UP STAT, where adabasname is the name of the ADABAS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls ADABAS conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that ADABAS is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule.

Table 114 ADABAS Termination Started Rule Structure

Item	Description
Title	**ADABAS Termination Started**
Name	ADAN51
Table	ADABAS
Message	`ADAN51 adabasname OPER,TYPEIN:`*`xxxxxx`* `when` *`xxxxxx`* `is ABEND`

Table 114 ADABAS Termination Started Rule Structure

Item	Description
Message Description	*xxxxxx* indicates the last operator command issued to ADABAS. When this command is ABEND, the message indicates that ADABAS termination started.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	ADABAS received an ABEND command from the operator and the IOA environment is informed that ADABAS is no longer up.
Rule Actions	Deletes condition or date CTO-adabasname-UP STAT, where adabasname is the name of the ADABAS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls ADABAS conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that ADABAS is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule.

Table 115 ADABAS MPM*xx* Termination Completed Rule Structure

Item	Description
Title	**ADABAS MPM*xx* Termination Completed**
Name	**IEF404I**
Table	**ADABAS**
Message	`IEF404I jjj ENDED { – TIME=hh.mm.ss}` when the message is issued from job MPM*.

Table 115 ADABAS MPM*xx* Termination Completed Rule Structure

Item	Description
Message Description	Job jjj ended.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	When ADABAS job ends, the IOA environment is informed that ADABAS is down.
Rule Actions	■ Deletes condition or date CTO-adabasname-UP STAT, where adabasname is the name of the ADABAS job or started task. ■ Adds condition or date CTO-adabasname-DOWN STAT, where adabasname is the name of the ADABAS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls ADABAS conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that ADABAS is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule. The rule assumes that the name of each production ADABAS environment starts with MPM. This prefix may need to be changed (in the rule's ON statement) in order to match site conventions.

Table 116 Suppress S MPM*xx* if ADABAS Is Active Rule Structure (part 1 of 2)

Item	Description
Title	**Suppress S MPM*xx* if ADABAS Is Active**
Name	S MPM*xx*
Table	ADABAS
Command	`S MPMxx`
Command Description	This command starts ADABAS (and the rule suppresses the command).
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	IN !CTO-adabasname-DOWN STAT where adabasname is the name of the ADABAS job or started task
Global Variables	None.
Rule Logic	In general, if command S MPMxx is issued when ADABAS is already up, either the new started task is immediately shut down, or it waits until the currently running task goes down and then starts automatically. In either case, other rules in this solution can be triggered, causing erroneous updating of prerequisite conditions, thus incorrectly triggering jobs in CONTROL-M. This rule prevents the situations described above by suppressing the S MPMxx command if ADABAS is already up. In this way, issuance of the termination messages, or starting of the task without a message being sent, is prevented and rules are not incorrectly triggered. This rule serves as a sample rule. A similar rule must be created for each ADABAS environment in the site.
Rule Actions	■ Suppresses the command. ■ Sends a message to user U-SHIFTOPER indicating that command S MPMxx was suppressed.
Activating the Rule	Once scheduled, the rule becomes active when prerequisite condition or date CTO-adabasname-DOWN STAT is deleted (where adabasname is the name of the ADABAS job or started task).

Table 116 Suppress S MPM*xx* if ADABAS Is Active Rule Structure (part 2 of 2)

Item	Description
Recommended Mode or Category	During the testing period activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that ADABAS is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule. The rule must be duplicated for each ADABAS job or started task in the site. To adapt each rule, the ADABAS name in the rule's ON statement and IN condition must be changed.

Copy and Clean ADABAS Protection Log

The ADABAS protection log is a dataset that keeps track of ADABAS activity. Usually the log is kept in dual mode, which means that two datasets are alternately used by ADABAS. When the current protection log becomes full, ADABAS switches to the other dataset and issues a message. This solution intercepts this message and schedules a job in CONTROL-M to copy the full protection log to a backup dataset and clean it. This solution eliminates the necessity to code the ADABAS EXIT 2 routine, which is otherwise responsible for cleaning the protection log.

Rules

The Copy and Clean ADABAS Protection Log solution includes the ADABAS Protection Log Is Full rule.

Rules Structure

The following tables describe the structures of the Copy and Clean ADABAS Protection Log solution rules.

Table 117 ADABAS Protection Log Is Full Rule Structure (part 1 of 2)

Title	**ADABAS Protection Log Is Full**
Name	ADAN21
Table	ADABAS
Message	`ADAN21 adabasname PROTECTION LOG DDPLOGRn started` when the message is issued from job MPM*.
Message Description	ADABAS is switching to the alternate protection log indicated by DDPLOGR*n*. The full protection log must be cleaned.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	As soon as the ADABAS protection log becomes full, a job is triggered to clean and copy it.
Rule Actions	Adds condition or date CTO-adabasname-LOG-GO STAT, where adabasname is the name of the ADABAS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.

Table 117 ADABAS Protection Log Is Full Rule Structure (part 2 of 2)

Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different automatic mechanism to clean the ADABAS log is already implemented, that mechanism must be removed before testing this rule. By automating ADABAS log dumping with CONTROL-O, the benefits of integrated operation are achieved. CONTROL-M tracks and controls the copy and clean job, and manages Quantitative resources used by the job. The rule assumes that the name of the production ADABAS environments starts with MPM. This prefix may need to be changed (in the rule's ON statement) in order to match site conventions. A CONTROL-M job scheduling definition and JCL for the ADABAS log copy and clean job must be created. The SOLVSCHD and SOLVJCL libraries contain a sample job scheduling definition and JCL to copy and clean ADABAS log. These samples can be adapted to the site's conventions and requirements. Separate job scheduling definitions must be created for each ADABAS production environment. The job must be defined as a cyclic job or cyclic started task (STC) with a MAXWAIT value of 99. It then only needs to be ordered once, but must not be removed manually from the CONTROL-M Active Jobs file. The MAXWAIT value of 99 ensures that the job is never removed from the Active Jobs file by the CONTROL-M New Day procedure. The cyclic job is always ready for submission. It is triggered by adding prerequisite condition or date CTO-*adabasname*-LOG-GO STAT (where *adabasname* is the name of the ADABAS job or started task). When execution of the job is completed, this condition is automatically deleted. This prevents cyclic re-invoking of the job and ensures that the job is only invoked again if the rule is triggered again.

Chapter 17

DMS

IDMS solutions are designed to keep track of the status of IDMS, to enable integration with CONTROL-M and to automate basic actions of cleaning the IDMS log and journal. Integration is done through IOA prerequisite conditions and is useful for scheduling jobs that use IDMS only when IDMS is active.

NOTE

Some messages (DO SHOUT actions) in the IDMS rules are sent to INCONTROL users (U-SYSDBA or U-SHFTOPER). Users with these names must be defined in the IOA Dynamic Destination table (CTMDEST). For details, see the considerations chapter of the INCONTROL for z/OS Administrator Guide.

Some rule definitions make use of the inverse IN condition feature. This feature activates rules only if the specified IN conditions are not set. For more information about using inverse IN conditions, see the section on SolveWare implementation considerations in Chapter 1, “Introduction.”

Solutions Provided

SolveWare subject IDMS contains the following solutions:

- IDMS Startup and Termination

  Monitors the IDMS startup and termination procedures, intercepts relevant messages and updates the appropriate IOA prerequisite conditions.

- Copy and Clean the IDMS Journal and Log

  Handles situations where the IDMS journal or log datasets become full. A copy and clean job is submitted automatically to prevent loss of information.

IDMS Startup and Termination

This solution handles IDMS startup and termination.

At sites operating both CONTROL-M and IDMS, the status of IDMS (up or down) impacts the start of jobs that run under CONTROL-M. Therefore, for the production IDMS in the system, the site must maintain two prerequisite conditions indicating IDMS status – CTO-IDMS-UP STAT and CTO-IDMS-DOWN STAT.

These two conditions are updated by CONTROL-O according to IDMS initialization and termination messages, and are used as prerequisite conditions for jobs run under CONTROL-M that require IDMS status information.

NOTE

While startup or termination is in process—but not yet completed—the IOA environment considers IDMS as both “not up” and “not down.” Therefore, IDMS not being up is not necessarily the same as IDMS being down, and vice versa.

Throughout this solution, IDMS C.V. stands for IDMS Central Version.

Rules

The IDMS Startup and Termination solution includes the following rules:

- IDMS C.V. Initialization Started
- IDMS C.V. Initialization Completed
- IDMS C.V. Termination Started
- IDMS C.V. Termination Completed
- Suppress S IDMS if IDMS Is Active

Rules Structure

The following tables describe the structures of the IDMS Startup and Termination solution rules.

Table 118 IDMS C.V. Initialization Started Rule Structure (part 1 of 2)

Item	Description
Title	IDMS C.V. Initialization Started
Name	IEF403I

Table 118 IDMS C.V. Initialization Started Rule Structure (part 2 of 2)

Item	Description
Table	IDMS
Message	`IEF403I jjj STARTED { - TIME=hh.mm.ss}` when the message is issued from job IDMS*.
Message Description	Job *jjj* starts.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	As soon as IDMS is started, the IOA environment is informed that IDMS is no longer down.
Rule Actions	Deletes condition or date CTO-*idmsname*-DOWN STAT, where *idmsname* is the name of the IDMS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls IDMS conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that IDMS is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule. The rule assumes that the name of the production IDMS jobs or started tasks starts with IDMS. This prefix may need to be changed (in the rule's ON statement) in order to match site conventions.

Table 119 IDMS C.V. Initialization Completed Rule Structure

Item	Description
Title	IDMS C.V. Initialization Completed
Name	IDMS DC013
Table	IDMS
Message	`IDMS DC013005 IDMS-CV/DC INITIALIZATION COMPLETE`
Message Description	IDMS completed initialization and is up.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	As soon as IDMS initialization is completed, the IOA environment is informed that IDMS is up.
Rule Actions	Adds condition or date CTO-*idmsname*-UP STAT, where *idmsname* is the name of the IDMS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls IDMS conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that IDMS is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule.

Table 120 IDMS C.V. Termination Started (part 1 of 2)Rule Structure

Item	Description
Title	IDMS C.V. Termination Started
Name	IDMS DC201
Table	IDMS

Table 120 IDMS C.V. Termination Started (part 2 of 2)Rule Structure

Item	Description
Message	`IDMS DC201002 IDMS CENTRAL VERSION QUIESCING`
Message Description	IDMS C.V. termination started.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	Intercepts IDMS termination message and informs the IOA environment that IDMS is no longer up.
Rule Actions	Deletes condition or date CTO-*idmsname*-UP STAT, where *idmsname* is the name of the IDMS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls IDMS conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that IDMS is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule.

Table 121 IDMS C.V. Termination Completed (part 1 of 2)Rule Structure

Item	Description
Title	IDMS C.V. Termination Completed
Name	IEF404I
Table	DMS
Message	`IDMS DC201003 IDMS CENTRAL VERSION SHUTDOWN or IEF404I jjj ENDED { – TIME=hh.mm.ss}` when the message is issued from job IDMS*.

Table 121 IDMS C.V. Termination Completed (part 2 of 2)Rule Structure

Item	Description
Message Description	IDMS DC201003 – IDMS C.V. shutdown successful. IEF404I – Job *jjj* ended.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	When IDMS terminates successfully or when the IDMS job ends, the IOA environment is informed that IDMS is down.
Rule Actions	■ Deletes condition or date CTO-*idmsname*-UP STAT, where *idmsname* is the name of the IDMS job or started task. ■ Adds condition or date CTO-*idmsname*-DOWN STAT, where *idmsname* is the name of the IDMS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls IDMS conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that IDMS is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule.

Table 122 Suppress S IDMS if IDMS Is Active Rule Structure (part 1 of 2)

Item	Description
Title	Suppress S IDMS if IDMS Is Active
Name	S IDMS
Table	IDMS
Command	S IDMS

Table 122 Suppress S IDMS if IDMS Is Active Rule Structure (part 2 of 2)

Item	Description
Command Description	This command starts IDMS (the rule suppresses the command).
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	IN !CTO-*idmsname*-DOWN STAT
Global Variables	None.
Rule Logic	In general, if command S IDMS is issued when IDMS is already up, the new started task is immediately shut down and normal termination messages are issued. These messages, however, give the appearance that the active IDMS has gone down and this may incorrectly trigger other rules in this solution and cause erroneous updating of prerequisite conditions, thus incorrectly triggering jobs in CONTROL-M. This rule prevents the situations described above by suppressing the S IDMS command if IDMS is already up. In this way, issuance of the termination messages is prevented and rules are not incorrectly triggered. This rule serves as a sample rule. A similar rule must be created for each IDMS environment in the site.
Rule Actions	■ Suppresses the command. ■ Sends a message to user U-SHIFTOPER indicating that command S IDMS was suppressed.
Activating the Rule	Once scheduled, the rule becomes active when prerequisite condition or date CTO-*idmsname*-DOWN STAT is deleted, where *idmsname* is the name of the IDMS job or started task.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	The rule must be duplicated for each IDMS job or started task in the site. To adapt each rule, the IDMS name in the rule's ON statement and IN condition must be changed. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that IDMS is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule.

Solution: Copy and Clean the IDMS Journal and Log

The IDMS journal and log are datasets that keep track of IDMS activity. The log must not be allowed to reach full capacity and must be cleaned when its utilization passes a certain threshold. The journal must be cleaned when it becomes full.

This solution intercepts messages regarding the utilization of the IDMS log and journal and schedules jobs in CONTROL-M to copy them to backup datasets and clean them when necessary.

Rules

The Copy and Clean the IDMS Journal and Log solution includes the following rules:

- IDMS Log Is *nn*% Full
- IDMS Journal Is Full

Rules Structure

The following tables describe the structures of the Copy and Clean the IDMS Journal and Log solution rules.

Table 123 IDMS Log Is *nn*% Full Rule Structure (part 1 of 3)

Item	Description
Title	IDMS Log Is *nn*% Full
Name	IDMS DC201
Table	IDMS
Message	`IDMS DC050001 DCLOG IS nn% FULL`
Message Description	IDMS log is *nn*% full.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.

Table 123 IDMS Log Is *nn*% Full Rule Structure (part 2 of 3)

Item	Description
Rule Logic	The rule is triggered when the IDMS log utilization reaches its threshold. If the percentage of Log utilization is higher than a predefined limit, the rule sets a condition to trigger a prescheduled job in CONTROL-M. The job copies the IDMS Log datasets. The job must be defined as a cyclic job or cyclic started task (STC). It is triggered by adding the prerequisite condition or date CTO-*idmsname*-LOG-GO STAT. For more details on the job scheduling definition, see the Customization item in this table.
Rule Actions	■ Extracts the IDMS log utilization percentage from the message text. ■ If utilization is over 80%, the following actions are performed: – Issues a command instructing CONTROL-O to write the Global variables. – Adds condition or date CTO-*idmsname*-LOG-GO STAT where *idmsname* is the name of the IDMS job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.

Table 123 IDMS Log Is *nn*% Full Rule Structure (part 3 of 3)

Item	Description
Recommended Mode or Category	During the testing period activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different automatic mechanism to clean the IDMS log is already implemented, the mechanism must be removed before testing this rule. By automating IDMS log cleaning with CONTROL-O, the benefits of integrated operation are achieved. CONTROL-M tracks and controls the copy and clean job and manages Quantitative resources used by the job. A CONTROL-M job scheduling definition and JCL for the IDMS log copy and clean job must be created. The SOLVSCHD and SOLVJCL libraries contain a sample job scheduling definition and JCL to copy and clean IDMS log. These samples can be adapted to a site's conventions and requirements. Separate job scheduling definitions must be created for each IDMS production environment. If the IDMS log tends to fill up either very quickly or very slowly, the threshold percentage (in the first SET statement) must be either lowered or raised respectively. The job must be defined as a cyclic job or cyclic started task (STC) with a MAXWAIT value of 99. It then only needs to be ordered once, but must not be removed manually from the CONTROL-M Active Jobs file. The MAXWAIT value of 99 ensures that the job is never removed from the Active Jobs file by the CONTROL-M New Day procedure. The cyclic job is always ready for submission. It is triggered by adding prerequisite condition or date CTO-*idmsname*-LOG-GO STAT (where *idmsname* is the name of the IDMS address space). When an execution of the job is completed, this condition is deleted. This prevents cyclic re-invoking of the job and ensures that the job is only invoked again if the rule is triggered again.

Table 124 IDMS Journal Is Full Rule Structure (part 1 of 3)

Item	Description
Title	IDMS Journal Is Full
Name	IDMS DC201
Table	IDMS
Message	`IDMS DC205003 TASK: idmsname; DISK JOURNAL IS FULL. SUBMIT IDMSAJNL JOB journalid`
Message Description	IDMS journal is full.

Table 124 IDMS Journal Is Full Rule Structure (part 2 of 3)

Item	Description
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%IDMS_JRNDUMP_*idmsname*.
Rule Logic	The rule is triggered as soon as the IDMS journal becomes full. It writes the journal's ID to a Global variable so that it can be used by the clean and copy job. The IDMS journal ID is indicated in the message. The Global AutoEdit variable is set to contain this ID and is referenced by the job JCL. The CONTROL-O $GLOBAL member is included in the job JCL by %%LIBSYM and %%MEMSYM AutoEdit control statements. The rule sets a condition to trigger a prescheduled job in CONTROL-M. The job copies the applicable IDMS journal dataset. The job must be defined as a cyclic job or cyclic started task (STC). It is triggered by adding the prerequisite condition or date CTO-*idmsname*-JRN-GO STAT. For more details on the job scheduling definition, see the Customization item in this table.
Rule Actions	■ Sets Global variable %%IDMS_JRNDUMP_*idmsname* to journalid indicated in the message. ■ Issues a command instructing CONTROL-O to write the Global variables. ■ Adds prerequisite condition or date CTO-*idmsname*-JRN-GO STAT.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.

Table 124 IDMS Journal Is Full Rule Structure (part 3 of 3)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different automatic mechanism to clean the IDMS journal is already implemented, the mechanism must be removed before testing this rule. By automating IDMS journal cleaning with CONTROL-O, the benefits of integrated operation are achieved. CONTROL-M tracks and controls the copy and clean job and manages Quantitative resources used by the job. A CONTROL-M job scheduling definition and JCL for the IDMS journal copy and clean job must be created. The SOLVSCHD and SOLVJCL libraries contain a sample job scheduling definition and JCL to copy and clean IDMS journal. These samples can be adapted to the site's conventions and requirements. Separate job scheduling definitions must be created for each production environment. The job must be defined as a cyclic job or cyclic started task (STC) with a MAXWAIT value of 99. It then only needs to be ordered once, but must not be removed manually from the CONTROL-M Active Jobs file. The MAXWAIT value of 99 ensures that the job is never removed from the Active Jobs file by the CONTROL-M New Day procedure. The cyclic job is always ready for submission. It is triggered by adding prerequisite condition or date CTO-*idmsname*-JRN-GO STAT (where *idmsname* is the name of the IDMS address space). When an execution of the job is completed, this condition is automatically deleted. This prevents cyclic re-invoking of the job and ensures that the job is only invoked again if the rule is triggered again.

Chapter 18

NETVIEW OPEN ACCESS

NETVIEW solutions are designed to automate several management aspects of NetView and to provide a bidirectional interface between CONTROL-O. The interaction and exchange between NetView and CONTROL-O enhances the power of both and saves the high costs of implementing console automation using NetView alone.

NOTE

Some messages (DO SHOUT actions) in the NETVIEW rules are sent to an INCONTROL user (U-SHFTOPER). A user with that name must be defined in the IOA Dynamic Destination table (CTMDEST). For details, see considerations chapter of the *INCONTROL for z/OS Installation Guide*.

Some rule definitions make use of the inverse IN condition feature. This feature activates rules only if the specified IN conditions are not set. For more information about using inverse IN conditions, see the section on SolveWare initialization considerations in Chapter 1, “Introduction.”

Solutions Provided

SolveWare subject NETVIEW contains the following solutions:

- NetView Startup and Termination

  Provides an easy procedure of starting and stopping the NetView started tasks. Monitors the NetView startup and termination processes, intercepts related messages and updates the appropriate IOA prerequisite conditions and global variables.

- Direct Access to NetView

  Provides a set of basic intercommunication facilities between CONTROL-O and NetView, which include exchange of variables and information, triggering of automation processes in both products and alert management.

NetView Startup and Termination

This solution handles NetView startup and termination and provides simple procedures for starting and stopping NetView. At sites operating both CONTROL-O and NetView, the status of NetView (up or down) impacts automation tasks and capabilities. Therefore, the site must maintain two prerequisite conditions for each task started by NetView, which indicate NetView status:

```
CTO-netview-UP 0101 and CTO-netview-DOWN STAT
CTO-netvssi-UP 0101 and CTO-netvssi-DOWN STAT
```

These conditions are updated by CONTROL-O according to the NetView started task initialization and termination messages and can be used as prerequisite conditions for CONTROL-O rules that use or control NetView resources.

This solution also maintains a set of global variables that indicate the availability of different NetView resources:

```
%%STATUS_netview
%%STATUS_netvssi
%%STATUS_NPDA
%%STATUS_NLDM
```

NOTE

While startup or termination is in process—but not yet completed—the IOA environment considers NetView or the NetView subsystem as both “not up” and “not down.” Therefore, NetView not being up is not necessarily the same as NetView being down, and vice versa.

Rules

The NetView Startup and Termination solution includes the following rules:

- NetView Subsystem Initialization Started
- NetView Subsystem Initialization Completed
- NetView Subsystem Termination Started
- NetView Subsystem Termination Completed
- NetView Initialization Started
- NetView Initialization Completed
- NetView Termination Completed
- Suppress S NETVSSI if NetView Subsystem is Active
- Suppress S NETVIEW if NetView Is Active
- Start NETVIEW and NETVSSI
- Stop NETVIEW and NETVSSI

Rules Structure

The following tables describe the structures of the NetView Startup and Termination solution rules.

Table 125 NetView Subsystem Initialization Started (part 1 of 2)Rule Structure

Item	Description
Title	NetView Subsystem Initialization Started
Name	IEF403I
Table	NETVIEW
Message	`IEF403I jjj STARTED { - TIME=hh.mm.ss}` when the message is issued from job NETVSSI.
Message Description	Job *jjj* starts.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	As soon as the NetView subsystem is started, the IOA environment is informed that the NetView subsystem is no longer down.
Rule Actions	Deletes condition or date CTO-*netvssiname*-DOWN STAT, where *netvssiname* is the name of the NetView subsystem job or started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.

Table 125 NetView Subsystem Initialization Started (part 2 of 2)Rule Structure

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls NetView conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that the NetView subsystem is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule. The rule assumes that the name of the production NetView subsystem environment is NETVSSI. This name may need to be changed (in the rule's ON statement) to match site conventions.

Table 126 NetView Subsystem Initialization Completed (part 1 of 2)Rule Structure

Item	Description
Title	NetView Subsystem Initialization Completed
Name	CNM541I
Table	NETVIEW
Message	`CNM541I NETVIEW SUBSYSTEM INITIALIZED SUCCESSFULLY`
Message Description	The NetView subsystem has completed initialization and is up.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%STATUS_*netvssiname* Status (UP or DOWN) of the NetView subsystem, where *netvssiname* is the name of the NetView subsystem job or started task.
Rule Logic	Intercepts a message that indicates the NetView subsystem has completed initialization, and informs the IOA environment that the NetView subsystem is up.

Table 126 NetView Subsystem Initialization Completed (part 2 of 2)Rule Structure

Item	Description
Rule Actions	■ Adds condition or date CTO-*netvssiname*-UP STAT (where *netvssiname* is the name of the NetView subsystem job or started task). ■ Sets Global variable %%STATUS_*netvssiname* to UP.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls NetView conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that the NetView subsystem is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule.

Table 127 NetView Subsystem Termination Started Rule Structure

Item	Description
Title	NetView Subsystem Termination Started
Name	CNM580I
Table	NETVIEW
Message	`CNM580I NETVIEW SUBSYSTEM ADDRESS SPACE IS TERMINATING`
Message Description	The NetView subsystem address space has received an MVS STOP command and is terminating.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%STATUS_*netvssiname* Status (UP or DOWN) of the NetView subsystem, where *netvssiname* is the name of the NetView subsystem job or started task.

Table 127 NetView Subsystem Termination Started Rule Structure

Item	Description
Rule Logic	Intercepts the NetView subsystem termination message and informs the IOA environment that the NetView subsystem is no longer up.
Rule Actions	■ Deletes condition or date CTO-*netvssiname*-UP STAT (where *netvssiname* is the name of the NetView subsystem job or started task). ■ Sets Global variable %%STATUS_*netvssiname* to DOWN.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism to control NetView conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that the NetView subsystem is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule.

Table 128 NetView Subsystem Termination Completed Rule Structure (part 1 of 2)

Item	Description
Title	NetView Subsystem Termination Completed
Name	IEF404I
Table	NETVIEW

Message	Either of the following messages:
	
	`IEF404I jjj ENDED { - TIME=hh.mm.ss}`
	
	`IEF450I jjj {ppp} sss - ABEND {Scde|Ucde} REASON=xxxxxxxx TIME=hh.mm.ss`
	
	when the message is issued from job NETVSSI.

Message Description	Job *jjj* ended or step *sss* of job *jjj* abended.

Table 128 NetView Subsystem Termination Completed Rule Structure (part 2 of 2)

Item	Description
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%STATUS_*netvssiname* Status (UP or DOWN) of the NetView subsystem, where *netvssiname* is the name of the NetView subsystem job or started task.
Rule Logic	When the NetView subsystem job or started task ends, the IOA environment is informed that the NetView subsystem is down.
Rule Actions	■ Adds condition or date CTO-*netvssiname*-DOWN STAT (where *netvssiname* is the name of the NetView subsystem job or started task). ■ Deletes condition or date CTO-*netvssiname*-UP STAT (where *netvssiname* is the name of the NetView subsystem job or started task). ■ Sets Global variable %%STATUS_*netvssiname* to DOWN.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls NetView conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that NetView is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule. The rule assumes that the name of the production NetView subsystem environment is NETVSSI. This name may need to be changed (in the rule's ON statement) to match site conventions.

Table 129 NetView Initialization Started Rule Structure

Item	Description
Title	NetView Initialization Started
Name	IEF403I
Table	NETVIEW
Message	`IEF403I jjj STARTED { - TIME=hh.mm.ss}` when the message is issued from job NETVIEW.
Message Description	Job *jjj* starts.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	As soon as NetView is started, the IOA environment is informed that NetView is no longer down.
Rule Actions	Deletes condition or date CTO-*netviewname*-DOWN STAT (where *netviewname* is the name of the NetView job or started task).
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls NetView conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that NetView is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule. The rule assumes that the name of the production NetView environment is NETVIEW. This name may need to be changed (in the rule's ON statement) to match site conventions.

Table 130 NetView Initialization Completed Rule Structure

Item	Description
Title	NetView Initialization Completed
Name	DSI112I
Table	NETVIEW
Message	`DSI112I NCCF READY FOR LOGONS AND SYSTEM OPERATOR COMMANDS`
Message Description	NetView completed initialization and is up.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%STATUS_*netviewname* Status (UP or DOWN) of the NetView subsystem (where *netviewname* is the name of the NetView subsystem job or started task).
Rule Logic	Intercepts a message indicating that NetView is ready for action and informs the IOA environment that NetView is up.
Rule Actions	■ Adds condition or date CTO-*netviewname*-UP STAT (where *netviewname* is the name of the NetView job or started task). ■ Sets Global variable %%STATUS_*netviewname* to UP.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls NetView conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. If the IOA environment already contains a prerequisite condition indicating that NetView is up or down (for example, if such a condition is already referenced by a CONTROL-M job scheduling definition), and if the prerequisite condition has a name that differs from the name indicated in this rule, then change the name of the prerequisite condition indicated in this rule to match your previous definition.

Table 131 NetView Termination Completed Rule Structure (part 1 of 2)

Item	Description
Title	NetView Termination Completed
Name	IEF404I
Table	NETVIEW

Message	Either of the following messages:
	
	`IEF404I jjj ENDED { - TIME=hh.mm.ss}`
	
	`IEF450I jjj {ppp} sss - ABEND {Scde|Ucde} REASON=xxxxxxxx TIME=hh.mm.ss`
	
	when the message is issued from job NETVIEW.

Message Description	Job *jjj* ended or step *sss* of job *jjj* abended.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Global Variables	`%%STATUS_`*`netviewname`*
	Status of the NetView job or started task, where *netviewname* is the name of the NetView job or started task.
	
	`%%STATUS_NPDA`
	Status of the NPDA.
	
	`%%STATUS_NLDM`
	Status of the NLDM.
	
	`%%NCCF_REPLY_`*`netviewname`*
	NCCF reply number, where *netviewname* is the name of the NetView job or started task.

Rule Logic	When NetView started task ends, the IOA environment is informed that NetView and its components are down.
Rule Actions	■ Adds condition or date CTO-*netviewname*-DOWN STAT, where *netviewname* is the name of the NetView job or started task. ■ Deletes condition or date CTO-*netviewname*-UP STAT. ■ Sets Global variable %%STATUS_*netviewname* to DOWN. ■ Sets Global variable %%STATUS_NPDA to DOWN. ■ Sets Global variable %%STATUS_NLDM to DOWN. ■ Sets Global variable %%NCCF_REPLY_*netviewname* to %%UNDEF.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.

Table 131 NetView Termination Completed Rule Structure (part 2 of 2)

Item	Description
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	If a different mechanism that controls NetView conditions (utility IOACND, CONTROL-M Event Manager, etc.) is already implemented, the mechanism must be removed before implementing this rule. Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that NetView is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule. The rule assumes that the name of the production NetView environment is NETVIEW. This name may need to be changed (in the rule's ON statement) to match site conventions.

Table 132 Suppress S NETVIEW if the NetView Subsystem is Active Rule Structure (part 1 of 2)

Item	Description
Title	Suppress S NETVSSI if the NetView Subsystem Is Active
Name	S NETVSSI
Table	NETVIEW
Command	`S NETVSSI`
Command Description	This command starts the NetView subsystem (and the rule suppresses the command).
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	IN !CTO-*netvssiname*-DOWN STAT where *netvssiname* is the name of the NetView subsystem job or started task.
Global Variables	None.

Table 132 Suppress S NETVIEW if the NetView Subsystem is Active Rule Structure (part 2 of 2)

Item	Description
Rule Logic	If command S NETVSSI is issued when the NetView subsystem is already up, the new started task is immediately shut down and normal termination messages are issued. These messages, however, give the appearance that the active NetView has gone down and this may incorrectly trigger other rules in this solution and cause erroneous updating of prerequisite conditions. This rule prevents the situation described above by suppressing the S NETVSSI command if the NetView subsystem is already up. In this way, issuance of the termination messages is prevented and rules are not incorrectly triggered. This rule serves as a sample rule. A similar rule must be created for each NetView subsystem environment in the site.
Rule Actions	■ Suppresses the command. ■ Sends a message to user U-SHFTOPER indicating that command S NETVSSI was suppressed.
Activating the Rule	Once scheduled, the rule becomes active when prerequisite condition or date CTO-*netvssiname*-DOWN STAT is deleted.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.

Table 133 Suppress S NETVIEW if NetView is Active Rule Structure (part 1 of 2)

Item	Description
Title	Suppress S NETVIEW if NetView is Active
Name	S NETVIEW
Table	NETVIEW
Command	`S NETVIEW`
Command Description	This command starts NetView (and the rule suppresses the command).
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	IN !CTO-*netviewname*-DOWN STAT Where *netviewname* is the name of the NetView job or started task.
Global Variables	None.

Table 133 Suppress S NETVIEW if NetView is Active Rule Structure (part 2 of 2)

Item	Description
Rule Logic	If command S NETVIEW is issued when NetView is already up, the new started task waits until the currently running task goes down and then starts automatically. This may affect other rules in this solution and cause erroneous updating of prerequisite conditions. This rule prevents the situation described above by suppressing the S NETVIEW command if the NetView subsystem is already up. In this way, the task does not start without a message being sent and rules are not incorrectly triggered. This rule serves as a sample rule. A similar rule must be created for each NetView subsystem environment in the site.
Rule Actions	■ Suppresses the command. ■ Sends a message to user U-SHFTOPER indicating that command S NETVIEW was suppressed.
Activating the Rule	Once scheduled, the rule becomes active when prerequisite condition or date CTO-*netviewname*-DOWN STAT is deleted.
Recommended Mode or Category	During the testing period activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	Change the name of the prerequisite condition indicated in this rule to match your previous definition if both of the following are true: ■ The IOA environment already contains a prerequisite condition indicating that NetView is up or down. For example, if such a condition is already referenced by a CONTROL-M job scheduling definition. ■ The prerequisite condition has a different name than the one indicated in this rule. The rule assumes that the name of the production NetView environment is NETVIEW. This name may need to be changed (in the rule's ON statement and IN condition) to match site conventions. The rule must be duplicated for each NetView subsystem in the site. To adapt each rule, the NetView subsystem name in the rule's ON statement and IN condition must be changed.

Table 134 Start NETVIEW and NETVSSI (part 1 of 2)Rule Structure

Item	Description
Title	Start NETVIEW and NETVSSI
Name	STRTNET

Table 134 Start NETVIEW and NETVSSI (part 2 of 2)Rule Structure

Item	Description
Table	NETVIEW
Command	STRTNET
Command Description	A user-defined command to start the NetView started tasks (NETVSSI and NETVIEW).
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%NETVSSI – Name of the NetView subsystem started task. %%NETVIEW = NETVIEW – Name of the NetView started task.
Rule Logic	Sets Global variables to fix the names of the NetView and NetView subsystem started tasks. Then issues START commands for the two started tasks whose names were specified above.
Rule Actions	■ Sets Global variable %%NETVSSI to NETVSSI. ■ Sets Global variable %%NETVIEW to NETVIEW. ■ Issues command S %%NETVSSI to start the NetView subsystem started task. ■ Issues command S %%NETVIEW to start the NetView started task.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	The rule sets the names of the NetView environments to NETVIEW and NETVSSI. These names may need to be changed (in the rule's DO SET statements) to match site conventions. The rule must be duplicated for each NetView. NetView started task name and NetView subsystem name in the rule's DO SET statements must be changed, as well as the names of the global variables different name must be given to each rule.

Table 135 Stop NETVIEW and NETVSSI Rule Structure (part 1 of 2)

Item	Description
Title	Stop NETVIEW and NETVSSI
Name	STOPNET
Table	NETVIEW
Command	STOPNET

Table 135 Stop NETVIEW and NETVSSI Rule Structure (part 2 of 2)

Item	Description
Command Description	A user-defined command to stop the NetView started tasks (NETVSSI and NETVIEW).
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%NETVSSI Name of the NetView subsystem started task. %%NETVIEW Name of the NetView started task. %%NCCF_REPLY Reply ID for issuing a command to NetView. %%STATUS_NETVIEW Status of NetView (UP or DOWN). %%STATUS_NETVSSI Status of the NetView subsystem (UP or DOWN).
Rule Logic	Checks the status of the NetView started tasks. If they are UP, terminates them. If they are DOWN, issues a message to notify that NetView is not up.
Rule Actions	■ Checks the status of NetView. If it is active, issues a CLOSE IMMED command to it. ■ Checks the status of the NetView subsystem. If it is active, issues an MVS STOP command to terminate it. ■ If none of the NetView started tasks was active, issues a message to INCONTROL user U-SHFTOPER to notify that NetView is not up.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	The rule must be duplicated for each NetView environment in the site. To adapt each rule, the names of the variables that contain the NetView started task and NetView subsystem names in the rule's DO statements must be changed. A different name must be given to each rule.

Direct Access to NetView

This solution handles intercommunication between CONTROL-O and NetView. At sites operating both NetView and CONTROL-O, integration between those two products is very important. It allows the automation procedures implemented with CONTROL-O to use any information available to NetView, without losing any automation benefits already acquired using NetView alone.

This solution provides rules and KOA scripts that enable transferring of Global variables and messages between NetView and CONTROL-O, issuing NetView commands from the IOA environment and handling of NPDA alerts.

Rules

The Direct Access to NetView solution includes the following rules:

- Save Reply ID of NCCF
- Send a Command to NCCF
- Set a Global Variable in NetView
- Get the Value of a Global Variable From NetView
- Set a Global Variable in CONTROL-O
- Record NPDA Alerts to the IOA Log
- Send Alert to NPDA

Rules Structure

The following tables describe the structures of the Direct Access to NetView solution rules, as well as the following KOA scripts:

- Manage a Preset NetView Environment
- Log On to NetView
- Get Value of NetView Variable
- Set Value of NetView Variable
- Enter NCCF Command and Get Response
- Enter a GENALERT Command to NetView
- Calculate Line Length in a NetView Screen
- Perform NetView GETSCREEN Commands

Table 136 Save Reply ID of NCCF Rule Structure

Item	Description
Title	Save Reply ID of NCCF
Name	DSI802I
Table	NETVIEW

Message	Either of the following messages:
	
	`DSI802I` *`domainid`* `REPLY WITH VALID NCCF SYSTEM OPERATOR COMMAND`
	
	`DSI803I` *`domainid`* `REPLY INVALID. REPLY WITH VALID NCCF TERMINAL OPERATOR COMMAND`

Message Description	A constant operator reply that prompts for a NetView command.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%NCCF_REPLY_*netviewname* NCCF reply number, where *netviewname* is the name of the NetView job or started task.
Rule Logic	When one of the above messages appears on the console, the message reply number is updated in a CONTROL-O Global variable. This variable is later accessed by another CONTROL-O rule that issues reply commands to NetView.
Rule Actions	Assigns the message reply number (system variable %%REPLY) to Global variable %%NCCF_REPLY_*netviewname*.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1—little or no customization is required.

Table 137 Send a Command to NCCF Rule Structure (part 1 of 2)

Item	Description
Title	Send a Command to NCCF
Name	NCCFCMD
Table	NETVIEW
Command	`NCCFCMD`
Command Description	A user-defined command to issue a NetView command through the NetView open reply message (WTOR).

Table 137 Send a Command to NCCF Rule Structure (part 2 of 2)

Item	Description
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%NCCF_REPLY_*netviewname* NCCF reply number, where *netviewname* is the name of the NetView job or started task.
Rule Logic	This command uses the NetView reply number, which is stored in a Global variable, to issue a NetView command. It waits for a reply from NetView and notifies the operator in case of an invalid command. The command is especially useful when a NetView command is to be issued from a source other than the console itself. For example, batch jobs, started tasks or CLISTs can use this command to issue NetView commands without having to determine the currently open NetView reply number.
Rule Actions	■ Using the NetView reply number (Global variable %%NCCF_REPLY_*netviewname*), issues the appropriate reply command. ■ Waits for NetView's reply and, if it announces an invalid command, notifies user U-SHFTOPER.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1—little or no customization is required.

Table 138 Set a Global Variable in NetView Rule Structure (part 1 of 2)

Item	Description
Title	Set a Global Variable in NetView
Name	SETNTVAR
Table	NETVIEW
Command	`SETNTVAR variable-name variable-value`
Command Description	A user-defined command to set a Global variable in NetView.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Table 138 Set a Global Variable in NetView Rule Structure (part 2 of 2)

Item	Description
Global Variables	None.
Rule Logic	This command invokes KOA script SETNVAR that sets a NetView Global variable. The command is very useful for integrating procedures of NetView and of CONTROL-O.
Rule Actions	■ Invokes KOA SETNVAR with the same parameters as the command. ■ Issues messages to INCONTROL user U-SHFTOPER according to the KOA's return code.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1—little or no customization is required.

Table 139 Get the Value of a Global Variable From NetView Rule Structure (part 1 of 2)

Item	Description
Title	Get the Value of a Global Variable From NetView
Name	GETNTVAR
Table	NETVIEW
Command	`GETNTVAR netview-variable CONTROL-O-variable`
Command Description	A user-defined command to get the value of a Global variable from NetView.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	The Global variable name is specified as the third word of the command.
Rule Logic	This command invokes KOA script GETNVAR that extracts the value of a NetView Global variable and assigns it to a Global variable in CONTROL-O. The command is very useful for integrating procedures between NetView and CONTROL-O.

Table 139 Get the Value of a Global Variable From NetView Rule Structure (part 2 of 2)

Item	Description
Rule Actions	■ Parses command text to extract required data. ■ Performs KOA script GETNVAR that uses the same variables as the command. ■ Issues messages to INCONTROL user U-SHFTOPER according to the KOA's return code.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1—little or no customization is required.

Table 140 Set a Global Variable in CONTROL-O Rule Structure (part 1 of 2)

Item	Description
Title	Set a Global Variable in CONTROL-O
Name	SETGLVAR
Table	NETVIEW
Command	`SETGLVAR variable-name variable-value`
Command Description	A user-defined command to set a Global variable in CONTROL-O.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	The Global variable name is passed as the second word of the command.
Rule Logic	This command sets a Global variable whose name is given by the first parameter and assigns the second parameter as its value. The command is very useful for integrating NetView with CONTROL-O. It enables NetView CLISTs or NetView exits to set Global variables in CONTROL-O using a simple operator command.
Rule Actions	Assigns the value of the second command parameter to a Global variable named by the first command parameter.

Table 140 Set a Global Variable in CONTROL-O Rule Structure (part 2 of 2)

Item	Description
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1—little or no customization is required.

Table 141 Record NPDA Alerts to the IOA Log Rule Structure

Item	Description
Title	Record NPDA Alerts to the IOA Log
Name	BNJ030I
Table	NETVIEW
Message	`BNJ030I operatorid time type ALERT RECEIVED FROM THE FOLLOWING RESOURCE: resource`
Message Description	An alert was generated in NetView.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	As soon as an alert is generated in NetView, a message is issued to the console. This message triggers this rule that sends it to IOA.
Rule Actions	Sends a message to user U-NETVALRT with the text of the alert.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1—little or no customization is required.
Customization	CLIST CTONVINI (supplied in the SOLVKOA library) issues a series of NPDA Set Recording Filter (SRFILTER) commands to enable alert generation and to allow CONTROL-O to receive NetView alert information. The CLIST must be adapted to site conventions and copied to one of the libraries specified in DD statement DSICLD of the NetView procedure. The NetView startup CLIST used at the site must be modified to issue command CTONVINI that sets the NPDA SRF filters.

Table 142 Send Alert to NPDA Rule Structure

Item	Description
Title	Send Alert to NPDA
Name	IEF450I
Table	NETVIEW
Message	`IEF450I jjj {ppp} sss - ABEND {Scde|Ucde} REASON=xxxxxxxx TIME=hh.mm.ss`
Message Description	Step *sss* of job jjj abended.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%NETVIEW_ALERT_JOB_*jobname* where *jobname* is the name of the abending job. Such a variable must be defined for each job that must be monitored by this rule.
Rule Logic	When a job abend message is intercepted, the rule checks if the job name appears in a predefined list of jobs. If the job's name appears on the list, an alert is sent to NetView about the abend.
Rule Actions	■ Checks if there exists a Global variable called %%NETVIEW_ALERT_JOB_*jobname*. ■ If such a variable exists, KOA script GENALRT is invoked to generate an alert in NetView.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 2—some customization is required before implementation.
Customization	The rule looks for a Global variable called %%NETVIEW_ALERT_JOB_*jobname*. To maintain a list of jobs that require generating an alert in NetView when they abend, the appropriate Global variables must be set by a rule that is activated during system or NetView startup. For additional information, see the Customization item in Table 148 on page 261.

Table 143 Manage a Preset NetView Environment Script Structure

Item	Description
Title	Manage a Preset NetView Environment
Name	NETVUE

Table 143 Manage a Preset NetView Environment Script Structure

Item	Description
KOA Script Description	This script manages the preset NetView environment that is used by the other KOA scripts and CONTROL-O rules in this solution.
Activating the KOA Script	Script NETVUE is activated automatically by a CONTROL-O server in response to requests for the preset NetView environment.
Parameters	Functions: INIT, TERM, RESET.
Global Variables	None.
KOA Script Logic	When the function is INIT, the KOA logs on to NETVIEW, types the userid and password and receives the main entry panel. The NetView screens are received by KOA script SCRNWAIT, which handles the special way in which NetView sends its screens (each part of the screen is sent separately). When the function is RESET, the KOA returns to the main NetView entry panel. When the function is TERM, the KOA logs off from NetView.
Recommended Category	The SolveWare category for this KOA script is 3—the script is provided as an example. Some customization is needed if the script is to be implemented.
Customization	The user ID, password and NetView VTAM application ID entered by this script must be modified according to site requirements. An alternative is to keep the user ID and password in a protected dataset and to extract them by using KOA file functions ALLOC, OPENFILE and GETFILE.

Table 144 Log On to NetView Script Structure (part 1 of 2)

Item	Description
Title	Log On to NetView
Name	LOGNET
KOA Script Description	This script is called by other scripts in order to log on to handle the online session with NetView. The script performs logon and logoff actions.
Activating the KOA Script	The KOA script is called using a CALLMEM statement from other KOA scripts (GETNTVAR, SETNTVAR, GENALERT).
Parameters	Functions: INIT, TERM, RESET
Global Variables	None.

Table 144 Log On to NetView Script Structure (part 2 of 2)

Item	Description
KOA Script Logic	When the function is INIT, the KOA logs on to NETVIEW, types the userid and password and receives the main entry panel. The NetView screens are received by KOA script SCRNWAIT, which handles the special way in which NetView sends its screens (each part of the screen is sent separately). When the function is RESET, the KOA returns to the main NetView entry panel. When the function is TERM, the KOA logs off from NetView.
Recommended Category	The SolveWare category for this KIOA script is 3—the script is provided as an example. Some customization is needed if the script is to be implemented.
Customization	The user ID, password, and NetView VTAM application ID entered by this script must be modified according to site requirements. An alternative is to keep the user ID and password in a protected dataset and to extract them by using KOA file functions ALLOC, OPENFILE, and GETFILE.

Table 145 Get Value of a NetView Variable Script Structure

Item	Description
Title	Get Value of a NetView Variable
Name	GETNVAR
KOA Script Description	This KOA script retrieves the value of a given NetView variable and sets a CONTROL-O Global variable to the obtained value.
Activating the KOA Script	The KOA script is activated by rule GETNTVAR in this solution.
Parameters	*netview-variable* CONTROL-O-*variable*.
Global Variables	CONTROL-O-*variable* specifies the name of the Global variable.
KOA Script Logic	The KOA script enters the NCCF facility and issues the supplied NetView CLIST GETCG using script NCCFCMD to display the value of the Netview variable. The CONTROL-O Global variable is set to the obtained value.
Recommended Category	The SolveWare category for this KOA script is 3—the script is provided as an example. Some customization is needed if the script is to be implemented.
Customization	CLIST GETCG (provided in the SOLVKOA library) must be copied to one of the NetView CLIST libraries specified in the DD statement DSICLD in the NetView procedure.

Table 146 Set Value of a NetView Variable Script Structure

Item	Description
Title	Set Value of a NetView Variable
Name	SETNVAR
KOA Script Description	This KOA script sets the value of a given NetView variable to the desired value.
Activating the KOA Script	The KOA script is activated by rule SETNTVAR in this solution.
Parameters	■ %A1 NetView variable to be set by the script ■ %A2 New value for the NetView variable
Global Variables	None.
KOA Script Logic	The KOA script enters the NCCF facility and issues the NetView command SETCGLOB using script NCCFCMD to set the value of the Netview variable.
Recommended Category	The SolveWare category for this KOA script is 1—little or no customization is needed.

Table 147 Enter NCCF Command and Get Response Script Structure

Item	Description
Title	Enter NCCF Command and Get Response
Name	NCCFCMD
KOA Script Description	This KOA script enters a given NCCF command and loads the response into CONTROL-O AutoEdit variables.
Activating the KOA Script	The KOA script is called using CALLMEM statements by other scripts in this solution.
Parameters	nccf-command.
Global Variables	None.
KOA Script Logic	The KOA script enters the NCCF facility, and issues command nccf-command, which is passed as a parameter. The responses are analyzed using script NEXTLINE and stored in CONTROL-O AutoEdit variables %%LINE1, %%LINE2,.....%%LINEn.
Recommended Category	The SolveWare category for this KOA script is 1—little or no customization is needed.

Table 148 Enter a GENALERT Command to NetView Script Structure (part 1 of 2)

Item	Description
Title	Enter a GENALERT Command to NetView
Name	GENALRT
KOA Script Description	This KOA script enters an alert to NetView using command GENALERT.
Activating the KOA Script	The KOA script is invoked by rule IEF450I in this solution.

Table 148 Enter a GENALERT Command to NetView Script Structure (part 2 of 2)

Item	Description
Parameters	■ %A1 System ID ■ %A2 Job name
Global Variables	None.
KOA Script Logic	The KOA script enters the NCCF facility and issues NetView command GENALERT to enter an alert to Netview.
Recommended Category	The SolveWare category for this KOA script is 3—the script is provided as an example. Some customization is needed if the script is to be implemented.
Customization	The resource hierarchy specified in the GENALERT command can be adapted to site conventions. The supplied hierarchy is CPU (smf ID), PROD (CTO) and PROG (jobname). CLIST CTONVINI (supplied in the SOLVKOA Library) issues a series of NPDA set recording filter (SRFILTER) commands to allow alert generation and to allow CONTROL-O to receive NetView alert information. The CLIST must be adapted to site conventions and copied to one of the libraries specified in DD statement DSICLD of the NetView procedure. The NetView startup CLIST used at the site must be modified to issue command CTONVINI, which sets the NPDA SRF filters.

Table 149 Calculate Line Length in a NetView Screen Script Structure

Item	Description
Title	Calculate Line Length in a NetView Screen
Name	NEXTLINE
KOA Script Description	This KOA script returns the number of the next line containing information on the NetView screen.
Activating the KOA Script	This script is activated automatically by another script that receives the NetView screen.
Parameters	■ %A1 The number of the current line. ■ %A2 The number of the first line containing information on the NetView screen. ■ %A3 The number of the last line on the screen that contains information.
Global Variables	None.
KOA Script Logic	The next line to be processed is returned to the script that invoked script NEXTLINE using variable %%nextline.
Recommended Category	SolveWare Category for this script is 1—minimal or no customization is required.

Table 150 Perform NetView GETSREEN Commands Script Structure

Item	Description
Title	Perform NetView GETSREEN Commands
Name	SCRNWAIT
KOA Script Description	Handles receipt of screens from NetView.
Activating the KOA Script	This script is called by other scripts that require access to a NetView screen.
Parameters	■ %A1 Maximum number of seconds to attempt to receive the screen. ■ %A2 Number of seconds after which, if the screen did not change, to consider the screen complete.
Global Variables	None.
KOA Script Logic	NetView screens are normally sent in several pieces. Script SCRNWAIT performs a NetView GETSCREEN command once per second and determines that the screen is complete when either no new screen data is received for a specified number of seconds (variable %A2), or a maximum number of seconds elapsed (variable %A1). One of the following return codes is returned by the script: ■ 0 – A complete screen was received. ■ 4 – Nothing was received. ■ 8 – A VTAM error occurred.
Recommended Category	SolveWare Category for this script is 1—minimal or no customization is required.

Chapter 19

CTORMT

Solutions in SolveWare subject CTORMT are designed to facilitate cross system automation communications. With the rules in this SolveWare solution, CONTROL-O rules can now issue commands to remote systems. Optionally, the rules can wait for the response to the remote command and analyze the response for further automation processing. CONTROL-O users can easily implement automation procedures that span multiple systems.

Solutions Provided

SolveWare subject CTORMT contains the Remote Operations solution. This solution provides an easy method for issuing commands to another system and optionally receiving the command response back in the local system.

Remote Operations

This solution provides an easy method for issuing commands to another system and optionally receiving the command response back in the local system. The response is provided in the rule's local variables and can be analyzed for further automation.

The solution requires CONTROL-O to be active and the CTORMT rules loaded, on all relevant systems (that is, the system on which the command originates and the system on which the command must be issued).

NJE or MAS connection between the systems is required.

NOTE

Some rules in this solution can be considered as internal rules. They do not interface with the user's rules and operations and they require no customization. These rules are marked "Internal" in the list below and no further documentation details are provided for them.

Rules

The Remote Operations solution includes the following rules:

- Issue Cross System Commands
- CTORMT Initialization
- Receive Remote Request (Internal)
- Pass Back Response (Internal)
- Suppress Remote Response From Console (Internal)

Rules Structure

The following tables describe the structures of the Remote Operations solution rules.

Table 151 Issue Cross-System Commands Rule Structure (part 1 of 2)

Item	Description
Title	Issue Cross-System Commands
Name	CTORMT
Table	CTORMT

Command	`CTORMT target-smfid`
	`WMODE=Y (Yes), N (No)`
	`[WRESP=Y (Yes), N (No)]`
	`[RMSG=response-message-ID]`
	`[TOUT=timeout-value]`
	`CMD=remote-command-text`

Command Description	Issue cross system commands according to rule arguments.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	■ %%RMT_ROUTE_*smfid* Command routing information. For more details, see customization instructions of rule RMTINIT in Table 152 on page 267. ■ %%CTORMT_CNT Remote command request counter. Used to assign a unique request ID to each request.
Rule Logic	The rule arguments are parsed. And the command is issued to the remote system according to these arguments. Response messages are stored in local variables in the originating rule.

Table 151 Issue Cross-System Commands Rule Structure (part 2 of 2)

Item	Description
Rule Actions	■ Parse rule arguments and perform validity checks. ■ If the command must be issued without waiting for a response, issue the command to the remote system and terminate. ■ If the rule is to issue the command and then wait to receive the response, do the following: – Obtain a unique request ID by increasing counter %%CTORMT_CNT. – Issue the command to CONTROL-O on the remote system, including request ID and response parameters. command is issued in command-response mode and the expected response message ID is CTORP<*rid*> where <*rid*> is the unique request ID. – Set local variable %%RLINE*n* to the *n*th response line. – Set local variable %%RLINES to the total number of response lines.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. SolveWare Category for this rule is 1—minimal or no customization is required.

Table 152 CTORMT Initialization Rule Structure (part 1 of 2)

Item	Description
Title	CTORMT Initialization
Name	RMTINIT
Table	CTORMT
Event	RMTINIT
Event Description	This Event rule initializes solution CTORMT.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%RMT_ROUTE_*smfid* Command routing information. For more details, see the Customization item in this table.

Table 152 CTORMT Initialization Rule Structure (part 2 of 2)

Item	Description
Rule Logic	This rule instructs the CTORMT solution how to communicate from one system to another in a given MVS complex. The rule is triggered following the loading of the CTORMT rule table to the CONTROL-O Active environment. Command routing information is stored in global variables according to the target system SMF ID. This information is then used by other rules for cross-system communication.
Rule Actions	For each system in the complex, sets global variable %%RMT_ROUTE_*smfid* to the appropriate command routing information.
Activating the Rule	The rule is activated once upon loading to the Active environment.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. Once you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare Category for this rule is 2—some customization is required before implementation.
Customization	In JES2 sites, routing variables (%%RMT_ROUTE_*smfid*) must contain the NJE node and (optionally) MAS member according to the syntax of JES2 command $N. For instance, MAS member 3 in NJE node 2 must have the value 2M3. In JES3 sites, these variables must contain system name according to the *SEND JES3 command.

Chapter 20

TCP/IP

SolveWare subject TCP/IP is designed to facilitate file transfers between various machines connected by a TCP/IP network. CONTROL-O rules are used to connect the systems and issue FTP commands to remote systems. Optionally, the rules can set a prerequisite condition to inform CONTROL-M of the file transfer.

Solutions Provided

SolveWare subject TCP/IP contains the File Transfer to Another System solution. This solution facilitates file transfer to another system and sets a prerequisite condition (to be referenced by CONTROL-M) upon successful completion of the transfer.

File Transfer to Another System

This solution facilitates file transfer to another system and sets a prerequisite condition (to be referenced by CONTROL-M) upon successful completion.

FTP software must be present on all systems to be accessed. This solution contains a REXX procedure used to interact with the FTP software. The CONTROL-O server used to run the REXX procedure in this solution must have a region size of at least two megabytes.

Rules

The File Transfer to Another System solution includes the Send a File using FTP rule.

Rules Structure

The following tables describe the structures of the File Transfer to Another System solution rules, as well as the following REXX procedure:

- FTP File "Put"

Table 153 Send a File using FTP Rule Structure (part 1 of 2)

Item	Description
Title	Send a File using FTP
Name	CTOFTPPT
Table	TCP/IP
DO RULE	CTOFTPPT *destination* *local-file* *foreign-file* *condition* *time-out*
DO RULE Description	The DO RULE statement that triggers this statement must contain the arguments shown above.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	None.
Rule Logic	The rule is triggered by a DO RULE statement in another rule. Rule arguments are parsed and a REXX procedure is invoked. (Rule arguments are described in detail in “Customization” below.) If a return code of 0 is returned from the REXX procedure, the specified condition is set. Otherwise, a message to the operator is issued.
Rule Actions	■ Invokes a REXX procedure to perform file transfer. ■ “Shouts” a message to the operator if the file transfer was not successful.
Activating the Rule	Once scheduled, the rule remains active until deleted from CONTROL-O.

Table 153 Send a File using FTP Rule Structure (part 2 of 2)

Item	Description
Recommended Mode or Category	During the testing period activate the rule in LOG mode. When you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. SolveWare Category for this rule is 1—minimal or no customization is required.
Customization	The following arguments must be specified in the DO RULE statement that invokes this rule: ■ *destination*–Name of the destination host (as defined to TCP/IP) ■ *local-file*–Name of the file to be transferred The file to be transferred can be either a sequential dataset or a member. ■ *foreign-file*–Name under which the file is to be stored in the destination host ■ *condition*–Name of the IOA condition to be set upon successful completion of file transfer Maximum length: 20 characters. ■ *time-out*–Value to be entered in the TIMEOUT field of the DO TSO statement that invokes REXX procedure CTOFTPPT

Table 154 FTP File "Put" REXX Procedure Structure (part 1 of 2)

Item	Description
Title	FTP File "Put"
Name	CTOFTPPT
REXX Script Description	This script opens a connection to the destination host system and transfers a specified file using that connection.
Activating the Script	This REXX procedure is activated using rule CTOFTPPT in this solution.

Table 154 FTP File "Put" REXX Procedure Structure (part 2 of 2)

Item	Description
Parameters	■ destination Name of the destination host (as defined to TCP/IP). ■ *local-file*–Name of the file to be transferred. The file to be transferred can be either a sequential dataset or a member. ■ *foreign-file*–Name under which the file is to be stored in the destination host. ■ *userid*–User ID of the job or started task or TSO under which the rule that activated this REXX procedure is running.
	Note: Parameters destination, *local-file* and *foreign-file* are passed as arguments from the rule that activated this REXX procedure. The value of parameter *userid* is extracted from AutoEdit variable %%$USERID.

Special Considerations	The script allocates a password file, named *userid*.NETRC (where *userid* is the value in %%$USERID as explained above), which is read by the FTP software. Since this file contains sensitive information, it must be protected by on-site security software.
	
	The structure of this file is:
	
	`MACHINE sys1 LOGIN usera PASSWORD passa`
	`MACHINE sys2 LOGIN userb PASSWORD passb`
	`MACHINE sys3 LOGIN userc PASSWORD passc`
	.
	.
	.
	
	where
	
	■ *sysn* is the host name (as defined to TCP/IP).
	■ *userx* is the user ID.
	■ *passx* is the password for the user (*userx* above).

Chapter 21

INITIATORS

SolveWare subject INITIATORS is designed to maximize initiator utilization by CONTROL-M. It adjusts IOA resources to reflect the actual state of initiators at your site.

Solution Provided

SolveWare subject INITIATORS contains the Quantitative Resource Handling solution. This solution updates the IOA Conditions file and the CONTROL-M Resources file according to events that affect initiator availability, such as a job start, or START, STOP, and SET commands.

Quantitative Resource Handling

This SolveWare solution uses Quantitative resources to track initiator availability. A Quantitative resource is created for each class monitored by CONTROL-M and these resources are updated whenever changes to initiator availability are detected. If, for example, a job in CONTROL-M is started in a specific class, CONTROL-M does not submit the job unless the relevant Quantitative resource indicates that an initiator that serves the appropriate class is available.

Normally (that is, without this SolveWare solution), Quantitative resources can determine resource availability based only on values defined within the IOA environment. Changes to initiators that were manually started, manually stopped, manually halted, or to initiators serving a manually submitted job (that is, a job not submitted by CONTROL-M) are not reflected in Quantitative resources.

The rules for this SolveWare solution monitor changes outside the IOA environment to initiators in the JES2 environment and adjust (that is, increase or decrease) relevant resource quantities accordingly.

Global variables that indicate initiator name, classes served by the initiator, counters and names of relevant Quantitative resources, are defined in rules INITFRST and INITCRT in this solution (see following sections). In multisystem configurations, CPU configuration is described using Global variables defined during SolveWare initialization by rule JES2GLBL in table INITSLV2. For more information, see the section on initializing SolveWare in Chapter 2, "STARTSYS."

Rules

The Quantitative Resource Handling solution includes the following rules:

- Initialize Environment, Start Monitoring Initiators
- Set Up Initiator Global Variables
- Perform DO RESOURCE Request
- Reset Variables at $TI, $SI and $ZI Commands
- JOB Started
- JOB Ended
- Initiator Ended

Rules Structure

The following tables describe the structures of the Quantitative Resource Handling solution rules.

Table 155 Initialize Environment, Start Monitoring Initiators Rule Structure (part 1 of 3)

Item	Description
Title	Initialize Environment, Start Monitoring Initiators
Name	INITFRST
Table	INITJES2
Event	INITFRST
Event Description	This Event rule initializes SolveWare solution INITIATORS. Global variables defined by this rule are necessary for the implementation of other rules in this solution.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Table 155 Initialize Environment, Start Monitoring Initiators Rule Structure (part 2 of 3)

Item	Description
Global Variables	■ %%CTM_CLASS_*index* Class to be monitored by CONTROL-M, indexed by a sequence number. For more information on this variable, see Customization in this table. ■ %%CTM Character used by CONTROL-M as a system identifier in a multisystem environment (AutoEdit variable %%$SIGN). %%ID_*smfid* Global variables are set by rule JES2GLBL in SolveWare subject STARTSYS. This rule determines the value for the %%ID_*smfid* variable for the current system and stores it in Global variable %%CTM. For example: If the SMFID of the current system is SYS1 and %%$SIGN=A was specified in the CONTROL-M installation on this machine, the following values are specified: %%ID_SYS1=A in rule JES2GLBL %%CTM=A in rule INITFRST ■ %%INIT_NUMBER Total number of initiators in the system. For example: %%INIT_NUMBER=15. ■ %%PROC_*index* Internal indicator of the status of Quantitative resources for classes served by a specific initiator. Valid values are – NO Quantitative resources relevant to the initiator are under construction. This value is specified at the beginning of the rule before variables relevant to the initiator have been reset. For example: %%PROC_1=NO – YES Quantitative resources relevant to the initiator reflect real initiator state. This value is specified when the variables relevant to this initiator have been reset . For example: %%PROC_1=YES
Rule Logic	This rule issues command $DI in command-response mode to generate a list of all initiators in the system. Each line of the response (message $HASP605) describes one initiator. AutoEdit variable %%INIT_NUMBER is set according to the number of lines in the command-response. This rule also sets Global variables that are used by other rules in this solution.

Table 155 Initialize Environment, Start Monitoring Initiators Rule Structure (part 3 of 3)

Item	Description
Rule Actions	■ Clears the environment from the previous run of this SolveWare solution. ■ Defines classes monitored by CONTROL-M. ■ Counts initiators in the system. ■ Initiates the environment for the solution.
Activating the Rule	The rule is triggered immediately once it has been ordered.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. When you are satisfied with the results of the rule and the log messages are no longer necessary, change the mode of this rule to PROD. The SolveWare category for this rule is 3—the rule is provided as an example. Implementation of the rule requires some customization.

Customization	Adapt the rule by changing the Global Variable list to reflect the real environment.
	
	Ensure that the appropriate number of %%CTM_CLASS_*index* variables are specified to define the classes to be monitored at your site.
	
	To avoid conflict with variables set by previous usage of this SolveWare, a variable with an index of one more than the number of classes must be specified with the value %%$UNDEF.
	
	For example:
	
	If classes A and B are monitored by CONTROL-M, the following must be specified:
	
	`%%CTM_CLASS_1=A,`
	`%%CTM_CLASS_2=B,`
	`%%CTM_CLASS_3=%%$UNDEF`
	
	In multisystem configurations, this rule uses Global variables defined by rule JES2GLBL. For customization information about these variables, see the section on initializing SolveWare in Chapter 2, “STARTSYS.”

Table 156 Set Up Initiator Global Variables Rule Structure (part 1 of 4)

Item	Description
Title	Set Up Initiator Global Variables
Name	INITCRT
Table	INITJES2
DO RULE	INITCRT
DO RULE Description	This rule sets values of Global variables referenced by other rules in this solution.

Table 156 Set Up Initiator Global Variables Rule Structure (part 2 of 4)

Item	Description
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	■ %%CLASS_*class*_T Total number of initiators in the system serving a specific class. For example: %%CLASS_A_T=25. ■ %%CLASS_*class*_O Total number of available initiators in the system (that is, in INACTIVE status) serving a specific class. For example: %%CLASS_A_O=15 ■ %%PROC_*index* Internal indicator of the status of Quantitative resources for classes served by a specific initiator. Valid values are – NO Quantitative resources relevant to the initiator are under construction. This value is specified at the beginning of the rule before variables relevant to the initiator have been reset. For example: %%PROC_1=NO – YES Quantitative resources relevant to the initiator reflect real initiator state. This value is specified when the variables relevant to this initiator have been reset. For example: %%PROC_1=YES ■ %%IN_PROCESS Internal indicator of the status of all Quantitative resources describing initiators. Valid values are – YES Quantitative resources relevant to all initiators are under construction. – NO Quantitative resources relevant to all initiators reflect actual initiator status.

Table 156 Set Up Initiator Global Variables Rule Structure (part 3 of 4)

Item	Description
	■ %%INIT_*index* Initiator name, indexed by sequence number. For example: %%INIT_1=1, %%INIT_10=A ■ %%JOB_*index* JOB ID of the job being processed by each initiator, indexed by sequence number. If an initiator is inactive, the %%JOB_index variable for that initiator has a value of NONE. For example: %%JOB_1=NONE %%JOB_10=JOB01384 ■ %%CLS_*index* List of classes served by a specific initiator, indexed by sequence number. For example: If the first initiator serves classes A, B and I and the second initiator serves only class A, the following is specified: – %%CLS_1=ABI – %%CLS_2=A ■ %%CLS_#_*index* Number of classes served by a specific initiator, indexed by sequence number. For example: If the first initiator serves classes A, B and I and the second initiator serves only class A, the following is specified: – %%CLS_#_1=3 – %%CLS_#_2=1 ■ %%STC_*index* STC number of a specific initiator, indexed by sequence number. The value of this variable for initiators that are unavailable (for example, initiators in HALTED or DRAINED status) is (NONE). For example: – %%STC_1=(NONE) – %%STC_10=STC32261).

Table 156 Set Up Initiator Global Variables Rule Structure (part 4 of 4)

Item	Description
Rule Logic	This rule is triggered by other rules in this solution: ■ At the start of processing. ■ Whenever certain JES2 commands are detected. This rule defines a Quantitative resource for each class monitored by CONTROL-M. This rule dynamically defines Global variables that describe CONTROL-M classes and initiator status. These variables are used by other rules in this solution to update Quantitative resources according to the number of available initiators serving each class.
Rule Actions	Initializes Global AutoEdit variables that describe initiators serving monitored classes. Sets all %%PROC_*index* Global variables to NO to prevent other rules in the solution from interrupting processing of this rule. Sets Global variable %%IN_PROCESS to YES to prevent other rules in this solution from interrupting processing of this rule. Issues command $DI in command-response mode to generate a list of all initiators in the system. Each initiator is described by three lines of the response message ($HASP605). AutoEdit variables %%INIT_*index*, %%JOB_*index*, %%CLS_#_*index*, %%CLS_*index*, %%STC_*index*, %%CLASS_*class*_T and %%CLASS_*class*_O are set according to information extracted from this message. When all response message lines for an initiator have been analyzed, the corresponding %%PROC_*index* Global variable is set to YES to re-enable update of relevant Quantitative resources. Sets Global variable %%IN_PROCESS to NO to again allow other rules in this solution to update Quantitative resources and Global variables relevant to JES2 initiator status. Calls rule STRS to update Quantitative resources.
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. When you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—this rule is provided as an example. Implementation of the rule requires some customization.
Customization	The rule uses Global variables defined by rule INITFRST in this solution. For customization information, see Table 155 on page 274.

Table 157 Perform DO RESOURCE Request Rule Structure

Item	Description
Title	Perform DO RESOURCE Request
Name	STRS
Table	INITJES2
DO RULE	STRS *resource-name resource-quantity*
Rule Description	This rule updates Quantitative resources to inform CONTROL-M of changes in initiator availability.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%CTM Character used by CONTROL-M as a system identifier in a multisystem environment. This variable is defined by the INITFRST rule in this solution (see Table 155 on page 274.) For example: %%CTM=B.
Rule Logic	The quantity specified for an Quantitative resource must be four digits in length. This rule determines the number of digits in the quantity to be specified and adds the appropriate number of leading zeroes to the quantity in the DO RESOURCE statement before updating the Quantitative resource.
Rule Actions	■ Builds the name of the Quantitative resource to be updated. The value stored in Global variable %%CTM is added as a suffix to this resource name to ensure that the resource name is unique. ■ Builds a 4-digit quantity for the Quantitative resource name from a received parameter. ■ Performs a DO RESOURCE action.
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. When you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—this rule is provided as an example. Implementation of the rule requires some customization.
Customization	The rule uses Global variables defined by rule INITFRST in this solution. For customization information, see Table 155 on page 274.

Table 158 Reset Variables at $TI, $SI, and $ZI Commands Rule Structure

Item	Description
Title	Reset Variables at $TI, $SI, and $ZI Commands
Name	$TI*
Table	INITJES2
Command	$TI, $SI, or $ZI These commands affect initiator status.
Command Description	■ $TI Reassigns one or more initiators to specified classes. ■ $SI Starts one or more initiators. ■ $ZI Halts one or more initiators.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	%%IN_PROCESS Internal indicator of the status of all Quantitative resources describing initiators. Valid values are ■ YES Quantitative resources relevant to all initiators are under construction. ■ NO Quantitative resources relevant to all initiators reflect actual initiator status.
Rule Logic	Each time command $TI, $SI or $ZI is issued, the rule triggers rule INITCRT of this solution to check initiator status and update relevant Global variables and Quantitative resources.
Rule Actions	■ Checks Global variable %%IN_PROCESS. If the value of this variable is YES, Quantitative resources relevant to all initiators are under construction and the rule waits until this construction is completed. ■ Triggers rule INITCRT to update all initiator related Quantitative resources.
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. When you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 1—little or no customization is required before implementation.

Table 159 Job Started Rule Structure (part 1 of 2)

Item	Description
Title	Job Started
Name	$HASP373
Table	INITJES2
Message	`$HASP373 jobname STARTED`
Message Description	This information message is issued when a job begins execution.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.
Global Variables	■ %%IN_PROCESS Internal indicator of the status of all Quantitative resources describing initiators. Valid values are – YES Quantitative resources relevant to all initiators are under construction. – NO Quantitative resources relevant to all initiators reflect actual initiator status. ■ %%PROC_*index* Internal indicator of the status of Quantitative resources for classes served by a specific initiator. Valid values are – NO Quantitative resources relevant to the initiator are under construction. This value is specified at the beginning of the rule before variables relevant to the initiator have been reset. For example: %%PROC_1=NO – YES Quantitative resources relevant to the initiator reflect real initiator state. This value is specified when the variables relevant to this initiator have been reset. For example: %%PROC_1=YES ■ %%JOB_*index* JOB ID of the job being processed by each initiator, indexed by sequence number. If an initiator is inactive, the %%JOB_*index* variable for that initiator has a value of NONE. For example: %%JOB_1=NONE %%JOB_10=JOB01384 ■ %%CLASS_*class*_O Total number of available initiators in the system (that is, in INACTIVE status) serving a specific class. For example: %%CLASS_A_O=15
Rule Logic	Each time a job is started, initiator related information is changed. This rule updates all relevant Global variables and Quantitative resources.

Table 159 Job Started Rule Structure (part 2 of 2)

Item	Description
Rule Actions	■ Determines the sequence number of the initiator serving started job. ■ Checks if Global variables and Quantitative resources relevant to the initiator serving the started job are under construction. ■ If Global variables and / or Quantitative resources relevant to this initiator are under construction, the rule does not update these variables. ■ Specifies the JOB ID of the started job in the %%JOB_index Global variable of the initiator processing the job. ■ Updates (decreases by one) the total number of available initiators (Global variables %%CLASS_*class*_O) for classes served by the initiator processing the started job. ■ Calls rule STRS to update relevant Quantitative resources.
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	LOG mode. When you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—this rule is provided as an example. Implementation of the rule requires some customization.

Table 160 Job Ended Rule Structure (part 1 of 3)

Item	Description
Title	Job Ended
Name	$HASP395
Table	INITJES2
Message	`$HASP395 jobname ENDED`
Message Description	This message is issued by JES2 when a job finishes executing.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Table 160 Job Ended Rule Structure (part 2 of 3)

Item	Description
Global Variables	■ %%IN_PROCESS Internal indicator of the status of all Quantitative resources describing initiators. Valid values are – YES Quantitative resources relevant to all initiators are under construction. – NO Quantitative resources relevant to all initiators reflect actual initiator status. ■ %%PROC_*index* Internal indicator of the status of Quantitative resources for classes served by a specific initiator. Valid values are – NO Quantitative resources relevant to the initiator are under construction. This value is specified at the beginning of the rule before variables relevant to the initiator have been reset. For example: %%PROC_1=NO – YES Quantitative resources relevant to the initiator reflect real initiator state. This value is specified when the variables relevant to this initiator have been reset. For example: %%PROC_1=YES ■ %%JOB_*index* JOB ID of the job being processed by each initiator, indexed by sequence number. If an initiator is inactive, the %%JOB_index variable for that initiator has a value of NONE. For example: %%JOB_1=NONE %%JOB_10=JOB01384 ■ %%CLASS_class_O Total number of available initiators in the system (that is, in INACTIVE status) serving a specific class. For example: %%CLASS_A_O=15
Rule Logic	Each time a job is ended, initiator related information is changed. This rule updates all relevant Global variables and Quantitative resources.
Rule Actions	■ Determines the sequence number of the initiator serving the job that ended. ■ Checks if Global variables and Quantitative resources relevant to the initiator serving the ended job are under construction. ■ If Global variables and / or Quantitative resources relevant to this specific initiator are under construction, the rule does not update these variables. ■ Specifies NONE in the %%JOB_*index* Global variable of the initiator that processed the ended job. ■ Updates (increases by one) the total number of available initiators (Global variables %%CLASS_*class*_O) for classes served by the initiator that processed the ended job. ■ Calls rule STRS to update relevant Quantitative resources.

Table 160 Job Ended Rule Structure (part 3 of 3)

Item	Description
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. When you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—this rule is provided as an example. Implementation of the rule requires some customization.

Table 161 Initiator Ended Rule Structure (part 1 of 2)

Item	Description
Title	Initiator Ended
Name	$HASP395
Table	INITJES2
Message	`$HASP395 INIT ENDED`
Message Description	This message is issued by JES2 when an initiator finishes executing.
Basic Scheduling Parameters	Always schedule this rule.
Runtime Scheduling Parameters	No special considerations.

Global Variables	■ %%JOB_*index*
	JOB ID of the job being processed by the initiator, indexed by sequence number. If an initiator is inactive, the %%JOB_index variable for that initiator has a value of NONE.
	For example:
	`%%JOB_1=NONE`
	`%%JOB_10=JOB01384`
	■ %%STC_*index*
	STC number of the initiator, indexed by sequence number. The value of this variable for initiators that are unavailable (for example, initiators in HALTED or DRAINED status) is (NONE).
	Examples
	– `%%STC_1=(NONE)`
	– `%%STC_10=STC32261`
	■ %%CLASS_*class*_T
	Total number of initiators in the system serving a specific class (for example, %%CLASS_A_T=25).
	■ %%CLASS_*class*_O
	Total number of available initiators in the system (that is, in INACTIVE status) serving a specific class (for example, %%CLASS_A_O=15).

Rule Logic	When an initiator is ended, the rule updates all Global variables related to that initiator.

Table 161 Initiator Ended Rule Structure (part 2 of 2)

Item	Description
Rule Actions	Determines the sequence number of the stopped initiator. Specifies NONE in the %%JOB_*index* Global variable of the stopped initiator. Specifies (NONE) in the %%STC_*index* Global variable of the stopped initiator. Updates (decreases by one) the total number of initiators and number of available initiators (Global variables %%CLASS_*class*_T and %%CLASS_*class*_O) for all classes served by the stopped initiator. Calls rule STRS to update relevant Quantitative resources.
Activating the Rule	Once ordered, the rule remains active until deleted from CONTROL-O.
Recommended Mode or Category	During the testing period, activate the rule in LOG mode. When you are satisfied with the results of the rule, change the mode to PROD to avoid log messages for the rule. The SolveWare category for this rule is 3—this rule is provided as an example. Implementation of the rule requires some customization.

Chapter 22

Cross-Reference Lists

This chapter includes the following cross-reference lists:

- Event Cross-Reference list
- Command Cross-Reference list
- Script Cross-Reference list
- Message Cross-Reference list

Event Cross-Reference List

The following is a list of events defined in CONTROL-O SolveWare solutions, the SolveWare table in which each event appears, and the page number where the event can be found.

Table 162 Event Cross Reference List (part 1 of 2)

Event	Table	Page
CHKDEVS	SHUTSYS	page 64
CHKINIT	SHUTSYS	page 66
CHKSTCS	SHUTSYS	page 67
CICSP1JC	CICS	page 170
DB2DOWN	DB2	page 178
DB2UP	DB2	page 177
DEVTYPES	DEVICES	page 88
INITFRST	INITJES2	page 274
JES2GLBL	INITSLV2	page 33
JOBOWNER	TESTJOB2	page 213
NEWCTO	SHUTSYS	page 68
ORDERALL	STARTSYS	page 42
RESDB2	DB2	page 183

Table 162 Event Cross Reference List (part 2 of 2)

Event	Table	Page
RESDUMP	DUMPDS	page 143
RESLOGRC	LOGREC	page 122
RESSMF	SMF	page 134
RMTINIT	CTORMT	page 267
SCTD	STARTSYS	page 41
SCTM	STARTSYS	page 41
SCTO	STARTSYS	page 40
SHUTCICS	SHUTSYS	page 55
SHUTCTD	SHUTSYS	page 51
SHUTCTM	SHUTSYS	page 52
SHUTDEVS	SHUTSYS	page 49
SHUTTSO	SHUTSYS	page 57
SHUTVTAM	SHUTSYS	page 62
STOPEXCP	OMEGAMON	page 200
STRTEXCP	OMEGAMON	page 199
SVTAM	STARTSYS	page 37

Command Cross-Reference List

The following is a list of commands defined in CONTROL-O SolveWare solutions, the SolveWare table in which each command appears, and the page number where the command can be found.

Table 163 Command Cross Reference List (part 1 of 2)

Command	Table	Page #
$TI*	INITJES2	page 281
BRODCAST	TSO	page 150
CTO*	DEVICES3	page 112
CTORMT	CTORMT	page 266
CTOREQ	DEVICES3	page 110
GETNTVAR	NETVIEW	page 255
IMSCMD-*	IMS	page 195
NCCFCMD	NETVIEW	page 253
S CICS	CICS	page 162
S IDMS	IDMS	page 230
S IMS	IMS	page 192

Table 163 Command Cross Reference List (part 2 of 2)

Command	Table	Page #
S MPMxx	ADABAS	page 221
S NETVIEW	NETVIEW	page 248
S NETVSSI	NETVIEW	page 247
SETGLVAR	NETVIEW	page 256
SETNTVAR	NETVIEW	page 254
SHUTSYS	SHUTSYS	page 47
STOPNET	NETVIEW	page 250
STRTNET	NETVIEW	page 249

Script Cross-Reference List

The following is a list of scripts defined in CONTROL-O SolveWare solutions and the page number where the script can be found.

Table 164 Script Cross Reference List

Script	Page #
BRODCAST	page 152
CTOFTPPT	page 271
GENALRT	page 261
GETNVAR	page 260
LOGNET	page 259
NCCFCMD	page 261
OMLEXSY	page 201
OMLOC	page 203
SETNVAR	page 261

Message Cross-Reference List

The following is a list of Messages defined in CONTROL-O SolveWare solutions, the SolveWare table in which each message appears, and the page number where the message can be found.

Table 165 Message Cross Reference List (part 1 of 6)

Message	Table	Page #
$HASP085	SHUTSYS	page 70
$HASP099	SHUTSYS	page 69
$HASP100	SUPJES2	page 77
$HASP101	SUPJES2	page 77
$HASP110	SUPJES2	page 77
$HASP111	SUPJES2	page 77
$HASP112	SUPJES2	page 77
$HASP113	SUPJES2	page 77
$HASP114	SUPJES2	page 77
$HASP115	SUPJES2	page 77
$HASP116	SUPJES2	page 77
$HASP117	SUPJES2	page 77
$HASP118	SUPJES2	page 77
$HASP125	SUPJES2	page 77
$HASP150	SUPJES2	page 77
$HASP160	SUPJES2	page 77
$HASP250	SUPJES2	page 77
$HASP301	SUPJES2	page 77
$HASP309	SUPJES2	page 77
$HASP310	CICS	page 161
$HASP310	IMS	page 191
$HASP373	INITJES2	page 282
$HASP373	INITJES2	page 77
$HASP395	INITJES2	page 283
$HASP395	STARTSYS	page 99
$HASP395	INITJES2	page 285
$HASP426	STARTSYS	page 36
$HASP523	SUPJES2	page 77
$HASP524	SUPJES2	page 77
$HASP532	SUPJES2	page 77
$HASP533	SUPJES2	page 77
$HASP534	SUPJES2	page 77

Table 165 Message Cross Reference List (part 2 of 6)

Message	Table	Page #
$HASP540	SUPJES2	page 77
ADAN01	ADABAS	page 218
ADAN21	ADABAS	page 223
ADAN51	ADABAS	page 218
ARC0100I	SUPSYS	page 82
ARC0200I	SUPSYS	page 82
ARC0208I	SUPSYS	page 82
ARC0503I	SUPSYS	page 82
ARC0728I	SUPSYS	page 83
ARC0734I	SUPSYS	page 83
BNJ030I	NETVIEW	page 257
CNM541I	NETVIEW	page 240
CNM580I	NETVIEW	page 241
CTM282I	TSO	page 152
CTO147I	INITSLV	page 32
DFH1588	CICS	page 164
DFH1701	CICS	page 159
DFH1799	CICS	page 160
DFH4583	CICS	page 166
DFHJC4583	CICS	page 166
DFHKE1799	CICS	page 160
DFHSI1517	CICS	page 157
DFHSI1588D	CICS	page 164
DFHTM1701	CICS	page 159
DFHTM1703	CICS	page 159
DFHTM1715	CICS	page 159
DFS3257I	IMS	page 188
DFS994I	IMS	page 187
DFS996I	IMS	page 194
DSI112I	NETVIEW	page 245
DSI802I	NETVIEW	page 253
DSI803I	NETVIEW	page 253
DSN3100I	DB2	page 174
DSN3104I	DB2	page 174
DSN3106I	DB2	page 174
DSN9022I	DB2	page 173
DSNC011I	DB2	page 180
DSNU086I	DB2	page 182

Table 165 Message Cross Reference List (part 3 of 6)

Message	Table	Page #
IAT2000	SUPJES3	page 79
IAT2006	SUPJES3	page 79
IAT5200	SUPJES3	page 79
IAT6101	SUPJES3	page 79
IAT6108	SUPJES3	page 79
IAT6118	SUPJES3	page 79
IAT7001	SUPJES3	page 79
IAT7007	SUPJES3	page 79
IAT7120	SUPJES3	page 79
IAT7450	SUPJES3	page 79
IAT9123	SUPJES3	page 79
IAT9124	SUPJES3	page 79
IAT9127	SUPJES3	page 79
IAT9190	SUPJES3	page 79
IAT9191	SUPJES3	page 79
ICH408I	SUPSYS	page 82
IDMS DC013	IDMS	page 228
IDMS DC050	IDMS	page 232
IDMS DC201	IDMS	page 229
IDMS DC205	IDMS	page 234
IEA405E	CONSOLE	page 146
IEA848I	SUPMVS	page 74
IEA911E	DUMPDS	page 137
IEA989I	SUPMVS	page 74
IEA994A	DUMPDS	page 137
IEA994E	DUMPDS	page 137
IEA995I	SUPMVS	page 74
IEC070I	SUPMVS	page 74
IEC130I	SUPMVS	page 74
IEC141I	SUPMVS	page 74
IEC161I	SUPMVS	page 74
IEC331I	SUPMVS	page 74
IEC501A	DEVICES	page 95
IEC501E	DEVICES	page 95
IEC502E	DEVICES	page 97
IEC705I	SUPMVS	page 74
IEC999I	SUPMVS	page 75
IEE043I	SUPMVS	page 75

Table 165 Message Cross Reference List (part 4 of 6)

Message	Table	Page #
IEE302I	DEVICES	page 92
IEE361I	SMF	page 132
IEE362A	SMF	page 125
IEE363I	SMF	page 129
IEE364I	SMF	page 130
IEE365I	SMF	page 131
IEE366I	SMF	page 125
IEE400I	SUPMVS	page 75
IEE711I	DUMPDS	page 137
IEE794I	DEVICES	page 86
IEF097I	SUPMVS	page 75
IEF125I	SUPMVS	page 75
IEF126I	SUPMVS	page 75
IEF170I	SUPMVS	page 75
IEF188I	SUPMVS	page 75
IEF196I	SUPMVS	page 75
IEF202I	SUPMVS	page 75
IEF233A	DEVICES	page 95
IEF233D	DEVICES	page 95
IEF234E	DEVICES	page 97
IEF236I	SUPMVS	page 75
IEF237I	SUPMVS	page 75
IEF238D	DEVICES	page 103
IEF238D	DEVICES2	page 105
IEF281I	DEVICES	page 93
IEF287E	DEVICES	page 108
IEF287I	SUPMVS	page 75
IEF288I	SUPMVS	page 75
IEF290E	DEVICES3	page 108
IEF403I	SUPMVS	page 239
IEF403I	ADABAS	page 217
IEF403I	CICS	page 157
IEF403I	IDMS	page 227
IEF403I	IMS	page 186
IEF403I	NETVIEW	page 244
IEF404I	SUPMVS	page 242
IEF404I	ADABAS	page 219
IEF404I	CICS	page 159

Table 165 Message Cross Reference List (part 5 of 6)

Message	Table	Page #
IEF404I	IDMS	page 229
IEF404I	IMS	page 190
IEF404I	NETVIEW	page 246
IEF404I	SHUTSYS	page 60
IEF450I	NETVIEW	page 258
IEF450I	TESTJOB1	page 206
IEF450I	TESTJOB2	page 209
IEF452I	SUPMVS	page 75
IEF452I	TESTJOB1	page 207
IEF452I	TESTJOB2	page 211
IEF453I	TESTJOB1	page 207
IEF453I	TESTJOB2	page 211
IEF453I	SUPMVS	page 75
IEF677I	SUPMVS	page 75
IEF722I	SUPMVS	page 75
IEF863I	OMEGAMON	page 203
IEF877E	DEVICES	page 108
IFB030I	LOGREC	page 117
IFB040I	LOGREC	page 115
IFB050I	LOGREC	page 118
IFB060E	LOGREC	page 115
IFB070I	LOGREC	page 118
IFB080E	LOGREC	page 115
IFB081I	LOGREC	page 115
IFB082I	LOGREC	page 117
IFB083I	LOGREC	page 118
IFB084I	LOGREC	page 118
IGF502E	DEVICES	page 102
IKJ144I	SUPSYS	page 81
IKJ572I	SUPSYS	page 81
IKJ605I	SUPSYS	page 81
IKJ606I	SUPSYS	page 81
IKT010D	SHUTSYS	page 59
IKT012D	SHUTSYS	page 60
IKT100I	SUPSYS	page 81
IKT108I	SUPSYS	page 81
IST093I	STARTSYS	page 38
IST234I	SUPSYS	page 81

Table 165 Message Cross Reference List (part 6 of 6)

Message	Table	Page #
IST259I	SUPSYS	page 81
IST530I	SUPSYS	page 81
IST532I	SUPSYS	page 81

Index

Symbols

A

B

C

D

E

F

G

I

J

K

L

M

N

O

P

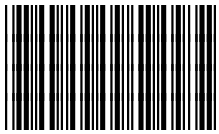
T

U

V

W

Notes



69909