CA VM:Schedule™

User Guide

Version 2.0, First Edition

9/25/2013

CA Technologies Product References

This document references the following CA Technologies products:

- CA Mainframe VM Product Manager

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Documentation Changes

CA VM:Schedule 2.0, First Edition, 9/25/2013

The following documentation updates have been made since the last release of this documentation:

- Global change—All references to the user ID for system administrator and maintenance functions changed from VMRMAINT to VMANAGER.
- Global change—Names and branding of related products were updated.
- Global change—Changed syntax diagrams to comply with latest CA standards.
- CHANGE Command (see page 70), QUERY Command (see page 101), and SCHEDULE Command (see page 112)—Added SYSNAME option.
- Running EXECs on CA VM:Schedule (EXEC): Restrictions (see page 44)—Added warning against using CMS WAKEUP.

Contents

Chapter 1: About This Book

The *CA VM:Schedule User Guide* presents concepts and procedures for tasks that end users typically perform. It also includes end user commands and the format for those commands.

This section contains the following topics:

Format Conventions for Code Syntax

This section describes the format and conventions used to document commands, utilities, and user exits. Each convention provides examples, describing how to use commands, how to use options, or how the system responds to user entries.

Note: The examples and instructions throughout this document use VMANAGER as the user ID for system administrator and maintenance functions. Also, this document uses VMSCHED as the user ID for the service virtual machine (svm). These user IDs are the default values. If you use non-default user IDs for the system administrator or the svm, replace the default values throughout this document with the values that you use.

Command Abbreviations

When a command contains uppercase and lowercase letters, then the uppercase letters denote the shortest acceptable abbreviation that you can use to type the command. However, when a code item appears entirely in uppercase letters, you cannot abbreviate the item.

You can type the code item in uppercase letters, lowercase letters, or any combination.

Example:

```
CMDNAme
```

In this example, you can enter CMDNA, CMDNAM, or CMDNAME in any combination of uppercase and lowercase letters.

Continuation

The code syntax or code fragment definitions can continue from one line to the next line. The following examples describe code continuation:

Example 1:

```
A | B C | D
```

This code is equivalent to the following code:

```
A
| B C
| D
```

Example 2:

```
{choice1 | choice2 | choice99}
```

This code is equivalent to the following code:

```
 { choice1
 | choice2
 | choice99 }
```

Default Values

An underlined code item denotes the default value. The system uses the default value unless you override it. You can override the default value by coding an option from the available list.

Example:

[parm1 | *parm2* | *parm3*]

In this example, the code item *parm1* is the default value, and this is used by the system when you do not specify any of the options. However, you can code *parm1*, *parm2*, or *parm3.*

Keywords and Constants

A keyword name or constant always appears in uppercase letters. Code the keyword name or constant exactly as shown in the following example:

Example:

STOP {*tracenumber* | * | [USER] *userid*}

This example displays the USER keyword.

Optional Choices

Defines optional code items—denoted by square brackets around a code item.

Example:

```
CMDName [parm1]
```

In this example, you can choose *parm1* or no parameter at all. However, when two or more items are enclosed in square brackets and separated by vertical bar characters, all of them are optional.

Multiple Optional Choices

When two or more items are enclosed in square brackets and separated by vertical bar characters, all of them are optional.

Example:

In this example, you can choose *parm1, parm2, parm3,* or nothing at all.

```
[parm1 | parm2 | parm3]
```

Positional Parameters

Commands with positional parameters are identified by nested square brackets. Each positional parameter requires the specification of all previous positional parameters. The following example describes the positional parameter:

Example:

```
CMDName [posparm1 [posparm2 [posparm3]]]
```

In this example, *posparm3* can be specified only when *posparm1* and *posparm2* are also specified.

Repeatable Choices

A list of code items enclosed in square brackets and followed by an ellipsis means that you can select more than one item or, in some cases, repeat a single item.

Example:

```
[value1 | value2 | value3] ...
```

In this example, you can choose a single value, more than one value, or none of the values.

Repetition

An ellipsis following a code item means that the code item can be repeated.

Example:

```
Repeat…
```

Required Choices

You must select one item from a list of items when they are enclosed in curly braces. The items are separated by a vertical bar character.

Example:

```
CMDNAme {A | B | C}
```

In this example, your choice results in CMDNAME A, CMDNAME B, or CMDNAME C.

Special Symbols

The following list describes the meaning of the special symbols used in codes:

- {} (encloses a list of operands, one of which is required).
- [] (enclose an optional operand or operands).
- " " (enclose the name of a syntax fragment)
- underline (identifies a default value)
- | (separates alternative operands)
- . . . (Indicates that the preceding item or group can be repeated).

Symbols

The following list displays symbols. These symbols should be coded exactly as they appear in the code syntax.

- * (Asterisk)
- : (Colon)
- , (Comma)
- = (Equal Sign)
- — (Hyphen)
- ( )(Parentheses)
- . (Period)

Syntax Fragments

Some codes use fragments, when the code syntax is too lengthy. The fragment name appears between double quotes in the code syntax.

The expanded fragment appears in the syntax after all other parameters or at the bottom of the code syntax. A heading with the fragment name identifies the expanded fragment.

Example:

```
CMDName "Parms"
Parms:

[A | B | C]
```

In this example, the fragment is named "Parms", and the expanded fragment appears at the bottom of the code syntax.

System Response

Uppercase characters represent system responses or prompts.

Example:

```
ENTER YOUR LOGON PASSWORD:
```

This example displays a system response.

User–Entered Commands or Records

User-entered commands are shown in lowercase letters even though you can enter commands in either upper or lower case.

Example:

```
vmsecure addentry writers tcom (noformat nowait
```

This example shows what a user-entered command looks like.

In this example "writers tcom" is the file name and file type of the directory entry you are adding.

However, if the entry is a record, it appears in uppercase letters.

Example:

```
ACCESS DRCT 1B0 U
```

This example shows a configuration record that is entered by a user.

Variables

Lowercase items in italics denote variables.

Example:

CMDNAme *varname*

In this example, *varname* represents a variable that you must specify when you code the command CMDNAME.

Related Documentation

Several guides that are associated with *CA VM:Schedule* make up the *CA VM:Schedule* library. Each guide addresses a different type of user depending on the tasks the user must perform. The *CA VM:Schedule* library includes the following books:

- *CA VM:Schedule User Guide* - presents concepts and procedures for tasks that end users typically perform. This guide also includes end-user commands and the format of those commands.
- *CA VM:Schedule Operator Guide* - presents concepts and procedures relevant to the day-to-day operation of *CA VM:Schedule*. The book also includes operator commands and the formats for those commands.
- *CA VM:Schedule Message Reference Guide* - provides the following information:
  - Lists all messages that *CA VM:Schedule* produces
  - Provides possible cause of the situation that generated the message, and any actions to take in response to the situation
  - Contains a cross-reference that lists the first 60 characters of each *CA VM:Schedule* message in alphabetical order.
- *CA VM:Schedule Administration Guide* - provides the following information:
  - Explanations of *CA VM:Schedule* and its administration
  - Instructions for customizing *CA VM:Schedule* to fit site-specific requirements
  - Descriptions of any special administrative operations that must be performed.
- *CA VM:Schedule Installation Guide* - gives information about installing and maintaining *CA VM:Schedule*.

Other CA Technologies product guides that are referenced in this book and to which you can refer include:

- *CA Mainframe VM Product Manager Reference Guide*
- *CA Mainframe VM Product Manager Interface Guide*
- *CA Mainframe VM Product Manager Generalized Report Writer Reference Guide*
- *CA VM:Batch User and Group Manager Guide*
- *CA VM:Batch Operator Guide*
- *CA VM:Batch Administration Guide*

Chapter 2: About CA VM:Schedule

CA VM:Schedule lets users schedule requests to process in the future instead of right now.

A scheduled request consists of an EXEC file, program, or command to be run and the scheduling instructions. Requests can print files, compile COBOL programs, issue CP and CMS commands, and run other types of CMS programs. You can schedule requests to run just once or to repeat at regular intervals. For example, you schedule the requests at 1:00 p.m. this coming Friday, 6:00 a.m. every business day, or on the last day of every month. You can schedule requests to run on your user ID or another user ID, or EXECs to run on the *CA VM:Schedule* service virtual machine.

Most scheduled requests run on your user ID. When it is time for the request to run, the server checks if you are logged on, and it sends you a reminder message to log off. Once you log off, the server autologs your user ID - that is, logs you on in disconnected mode. Your request then runs as if you entered the command yourself. The server monitors and records what happens during request processing. When your request runs for the last time, and whenever an error occurs within the request, the server sends a file to your reader to let you know.

You can set up frequently repeated work in advance, such as monthly accounting reports. You can schedule time-consuming tasks to run when you are away from the office so that the tasks do not interrupt your other work. You can also back up your spool files every day automatically after you log off, and remind colleagues about meetings. Programmers can compile and test programs overnight. Datacenter staff can automate system backups, performance reporting, and system control utilities.

This section contains the following topics:

Accessing and Leaving CA VM:Schedule

Before you use *CA VM:Schedule*, ensure that your PROFILE EXEC contains one of the following lines: SET AUTOREAD ON or SET RUN ON, or that the IPL statement in your directory entry contains AUTOCR.

Check your PROFILE EXEC and verify that it does *not* contain SET AUTOREAD OFF. If it does, *CA VM:Schedule cannot* execute your commands.

To use *CA VM:Schedule*, type **VMSCHED** from CMS and press Enter. When you first start the server, a copyright screen can appear. To clear the screen, press Enter or wait 15 seconds.

Note: VMSCHED is the default name of the *CA VM:Schedule* service virtual machine. If the *CA VM:Schedule* service virtual machine at your site has a different name, use that name instead.

After the copyright screen clears, the User Main Menu appears at whatever fluency level you have set for *CA VM:Schedule*. See the *Moving Through Screens and Menus* (see page 18) section for complete information on using *CA VM:Schedule* screens.

You can also use the *CA VM:Schedule* commands in line mode by entering the command from CMS. Complete formats for line-mode commands are listed in the *Command Reference* (see page 67) section.

To leave the server, press PF3 to return to the User Main Menu, and press PF3 again to return to CMS.

Moving Through Screens and Menus

To execute *CA VM:Schedule* user commands, access the product with authorization as a USER. You can access *CA VM:Schedule* in full-screen menus or in line mode. The screens are easier to use, especially if you are unfamiliar with *CA VM:Schedule* or you do not schedule requests often. The line-mode commands let you use *CA VM:Schedule* within EXECs or from a program.

Using Screens

To open the full-screen menu for user commands, type **vmsched user** from CMS and press Enter. *CA VM:Schedule* displays the Novice User Main Menu, as shown in the following example screen:

```
                          Novice User Main Menu                  VM:Schedule
--------------------------------------------------------------------------------
To choose one of the following, type its number and press ENTER.

     1  SCHEDULE  Schedule a request for initiation by VM:Schedule
                  to run on your userid.
     2  CANCEL    Cancel a scheduled request.
     3  SKIP      Skip the next initiation(s) of a scheduled request.
     4  QUERY     Display information about requests.
     5  WHEN      Display future initiation(s) of a request.
     6  SET       Control the way VM:Schedule prompts you.
     7  CHANGE    Change a request.




                Copyright (c) 2013 CA. All rights reserved.
--------------------------------------------------------------------------------
PF: 1  Help     2  ...      3  End       4  Return    5  ...       6  ...
PF: 7  ...      8  ...      9  ...      10  Print    11  ...      12  ...

===>
--------------------------------------------------------------------------------
```

Note: VMSCHED is the default name of the *CA VM:Schedule* service virtual machine. If the *CA VM:Schedule* service virtual machine at your site has a different name, use that name instead.

Type the number of the task you want to perform and press Enter. If you prefer, you can enter the command name or its abbreviation instead. *CA VM:Schedule* displays the fill-in-the-blank screen for that task. Where appropriate, default values for fields are shown. When you complete the task, press PF12 to have the server process your instructions. If you forget to fill in a required field, the server prompts you to enter the information.

If you want further explanation for any item on a screen, press Tab to put your cursor on the item and press PF1. *CA VM:Schedule* displays an explanation. Press PF3 to return to the task screen.

Fluency Levels

CA VM:Schedule offers screens for three fluency levels: NOVICE, FLUENT, and EXPERT. The levels differ in the number of selections available in the main menus, and in the number of scheduling options available from the SCHEDULE screens. The more advanced the level, the more things you can do.

NOVICE Level

The NOVICE level lets you perform basic scheduling tasks, such as scheduling simple requests, canceling and skipping requests, and displaying information about requests and future initiations of requests. The NOVICE level makes it easy for first-time users to schedule requests. On the NOVICE SCHEDULE screen, you can specify the following request parameters:

- The date and time a request is to start
- How often you want the request to repeat (hourly, daily, every business day, weekly, monthly, or quarterly)
- When you want the request to stop

FLUENT Level

The FLUENT level lets you perform intermediate scheduling tasks. In addition to the scheduling options available with the NOVICE level, the FLUENT SCHEDULE screens let you perform the following tasks:

- Start the requests on specific days of the week (for example, SUN or MON).
- Schedule run times spanning midnight.
- Specify shifts and ranges inside or outside of which requests are to run.
- Repeat the requests on specific days of the week (for example, SUN, MON), at the end of the month or quarter, or yearly.
- Put the requests on hold.
- Display system configuration information.
- Specify how long the server retries to run a request that cannot start at its scheduled time.

EXPERT Level

The EXPERT screens let you perform advanced scheduling tasks. In addition to the capabilities of the NOVICE and FLUENT screens, the EXPERT screens let you perform the following tasks:

- Schedule the requests to start on, or run until a particular day, week, or month of a given week, month, or quarter. For example, you can schedule a request to run on the first business day of the second quarter.
- Schedule the requests to start or stop on a particular day of the week (for example, SUN, MON).
- Schedule the requests by class.
- Schedule the requests to that another user releases.
- Schedule the EXECs to execute on the *CA VM:Schedule* service virtual machine.
- Repeat the requests on a particular day, week, or month of a given week, month, or quarter.
- Delay and release the requests.
- Specify resource limits for a request.
- Specify what monitoring to perform, if any.

Changing Fluency Levels

The default fluency level is NOVICE. You can change fluency levels by using the SET screen or the SET DISPLAY command. You can use the SET DISPLAY command from line mode to change levels before entering a *CA VM:Schedule* User main menu. For example, to change your fluency to FLUENT from line mode, enter the following command:

```
vmsched set display fluent
```

Once you change fluency levels, the new level remains in effect until you change levels again. See *SET DISPLAY Command* (see page 130) for information about using this command.

WARNING! If you define a request at one fluency level, and you later copy or update the request while using a lower level, you can lose some parts of the request. In particular, you can lose repeat options available only at higher fluency levels, request monitoring, logging, and resource allocation options. For example, suppose you define a request at the EXPERT level to repeat every other business day of the current quarter, and later copy it while using the FLUENT level. Because you cannot repeat requests this way from the FLUENT level, the copied request will not specify any repeat initiations. The request will use the default repeat setting, which is to not repeat the request.

Copying Requests

You can build a new request that is based on an existing one by using the PF5 (Copy) key on the SCHEDULE or EXEC screen. In the Request name field, enter the name of the request you want to copy and press PF5. The server fills in the fields defining the request automatically, leaving the Request name and user ID fields blank. Enter the name of the new request (and new user ID if you want). You can also change other fields now. When you complete the changes, press PF12 (Submit) to submit the request.

Changing Requests

You can change an existing request by selecting CHANGE from any level User Main Menu or using the CHANGE line-mode command.

Note: For information about using the CHANGE line-mode command, see *CHANGE Command (see page 70).* For information about using the full-screen CHANGE command, see *Changing Scheduling Parameters (see page 63).*

Scheduling Example

In this example, you want to schedule the REPORT request to run the REPORT command for your Accounts Payable department every day at 7 a.m. You can perform this task from the Novice user main menu. On this menu, select option 1, Schedule. *CA VM:Schedule* displays the novice User Schedule screen.

```
                              Novice User Schedule                   VM:Schedule
--------------------------------------------------------------------------------

    Request name: Report__     (Your logon password:          )
    Command to execute:  REPORT ACCT_____________________________________________
                         ____________________________________________________

                                FIRST RUN OPTIONS

    START at: 07:00:00 (hh:mm:ss)              From: 03/19/yy (mm/dd/yy)

                                REPEAT OPTIONS

   Run every          Hour, Day, Business Day, Week, Month,
                      Quarter    (H/D/B/W/M/Q): D

                                FINAL RUN OPTIONS

  Run the request until: __/__/__ (mm/dd/yy date of last run)
--------------------------------------------------------------------------------
PF: 1  Help     2  ...      3  End       4  Return    5  Copy      6  ...
PF: 7  ...      8  ...      9  ...      10  Print     11  ...      12  Submit
0185I ENTER DATA AND PRESS 'ENTER'.
===>
--------------------------------------------------------------------------------
```

Here, you only have to fill in the following fields to enter the request:

- Request name specifies the name that you give the report (REPORT)
- Command specifies the name of the command, including parameters, to be executed (REPORT ACCT)
- START at: indicates the time of day that you want the request to run (07:00)
- Run every indicates how often you want to run the request (D for daily).

Using CA VM:Schedule Line-Mode Commands

To use line-mode commands, enter the name of the service virtual machine (default is **VMSCHED**) followed by the command name and any operands. When a line mode command completes, a return code is presented to the issuing user.

For example, to transfer the ACCOUNTS request from user ALICE to user LOUISE through a line-mode command, enter the following command from CMS:

```
vmsched transfer accounts alice louise
```

Help with CA VM:Schedule Commands

You can review explanations of *CA VM:Schedule* commands at your terminal. To get help for commands, enter the following command:

```
help vmsched menu
```

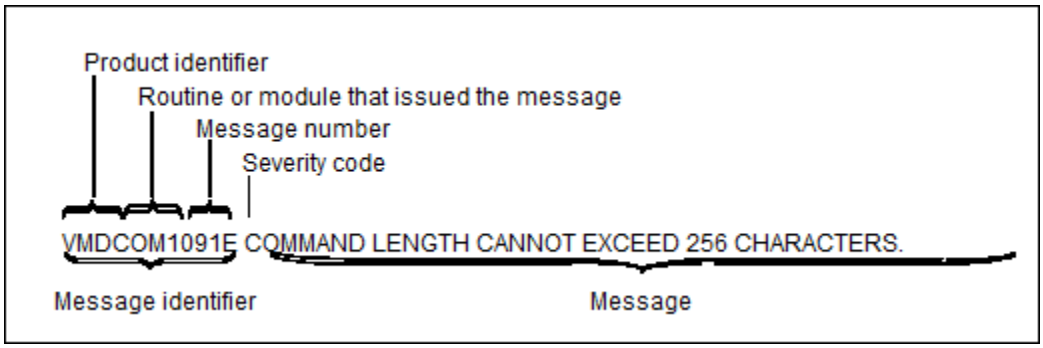
A help menu displays. Move the cursor to the command that you need help with and press Enter.

CA VM:Schedule Messages

As you use *CA VM:Schedule*, you receive messages that do one of the following:

- Inform you of the status and progress of the function or task
- Prompt you for information about your system
- Indicate if an error occurred
- Provide you instructions for resolving errors.

All messages are in the format that the following graphic shows:



Severity Codes

You can tell the type of message by its last letter, which is its *severity code*. The severity code can be one of the following codes:

Severity Code	Meaning
A - Action message	You need to perform an action, for example, move the cursor or enter information.

Severity Code	Meaning
E - Error message	The function or task can continue normally if you correct the error. The function of task can end.
I - Information message	*CA VM:Schedule* has completed or a *CA VM:Schedule* task has completed. An information message that appears immediately after a severe error message (severity code 5), it tells you how to correct the problem the severe error message identified.
R - Response message	*CA VM:Schedule* requests information. The function or task continues after you supply the required information.
S - Severe error message	An error that prevents the function or task from continuing has occurred. The product returns you to CMS. You must correct the error and restart the procedure.
T - Terminal error message	A serious internal error has occurred that prevents the *CA VM:Schedule* service virtual machine from continuing execution.
W - Warning message	An abnormal condition has occurred. The function or task continues, but the results may be affected by the abnormal condition

Help With Messages

The CMS HELP facility provides information about *CA VM:Schedule* commands and system messages. The information is the same as the information in this guide. The HELP files include complete descriptions of messages that are listed but not explained here.

The *CA VM:Schedule Messages Reference Guide* lists all messages that the server can generate, explains the possible causes of each message, and tells you how to respond to that message. For more information about a system message you receive, refer to *CA VM:Schedule Message Reference Guide* or use the CMS HELP facility for that message.

To use the CMS HELP facility for a *CA VM:Schedule* message, type **help** followed by a space, then the three-character product identifier (**vmd**), the message number, and the severity code. Do not type the three-letter code that displays after the product identifier; this code is the routine or module that issued the message.

For example, to get help on the message VMDCOM1091E, enter the following command from CMS:

```
help vmd1091e
```

When you have finished reading the message help, press PF3 to exit the CMS HELP facility.

For more information about using the CMS HELP facility, refer to the CMS user guide for your system.

Authorizations

To perform the tasks and use the commands that this guide describes, you must be authorized as a user in the *CA VM:Schedule* system. Contact your *CA VM:Schedule* system administrator if you experience difficulties.

Chapter 3: Basic Scheduling

This chapter describes basic scheduling tasks that you can accomplish with *CA VM:Schedule*. Each task illustrates examples. Most of the examples show the line-mode SCHEDULE command, however, some include a full-screen example.

See *SCHEDULE Command* (see page 112) for the complete command format and all options.

This section contains the following topics:

Running a Request Once

You can schedule requests to run any time in the future; five minutes from now, 6:00 a.m. tomorrow, next month, or the first of April next year. You can complete the following tasks with this feature:

- Process a large SCRIPT file during your lunch hour
- Remind co-workers about an upcoming meeting
- Run a large report at night, instead of waiting for it to finish while you could be working at your terminal.

Line-Mode Examples

Your boss just asked you to run a complex database query. The query, QUERY1 EXEC, takes a long time to run, relies on data that is updated regularly, and must be run after 5:00 p.m. You want to schedule this job, BOSSQRY, in a command (QUERY1 EXEC) to run at night after you leave work.

- To schedule the request BOSSQRY to run at 9:00 tonight, enter the following command:

  ```
  vmsched schedule bossqry query1 (at 21:00:00
  ```

- To schedule BOSSQRY to run at 9:00 p.m. this coming Monday, enter the following command:

  ```
  vmsched schedule bossqry query1 (at 21:00:00 on mon
  ```

- To schedule it to run at 9:00 p.m. on the first Monday in June, enter the following command:

  ```
  vmsched schedule bossqry query1 (at 21:00:00 from 1 mon jun
  ```

- To schedule it to run at 9:00 p.m. on June 30, enter the following command:

  ```
  vmsched schedule bossqry query1 (at 21:00:00 from 06/30/yy
  ```

At 9:00 p.m. (21:00:00 on the 24-hour clock) on the appropriate day, *CA VM:Schedule* autologs your user ID, and executes the command QUERY1.

The name you give a request (for example, *bossqry* in the example above) can contain up to eight characters. You cannot give two requests the same name if you schedule them from the same user ID.

The command line for the EXEC or command you schedule can be longer than a single word; for example, you could schedule the following command line:

```
sendfile query1 data a to theboss.
```

If the command line contains parentheses, omit the command line and let *CA VM:Schedule* prompt you for it. If the command line is longer than 108 characters and spaces, put it in an EXEC and schedule that EXEC instead.

Log File

You can direct the server to put the request results and a notice that your request BOSSQRY completed correctly in a log file named BOSSQRY VMSCHED, and to send this file to your reader. The server sends you a log file when your request runs for the last time automatically, therefore, always for a request run once. Also, if your request did not complete correctly, the server sends you a log file that indicates why the request did not run, unless you specify otherwise. For details about changing how often *CA VM:Schedule* sends you a log file, see the *Controlling the Request Log File* (see page 42) section.

Full-Screen Example

Using the same scenario as above, schedule your query, QUERY1 EXEC, to run at 9:00 p.m. on June 20, using the *CA VM:Schedule* Novice User Schedule screen. The following graphic shows how to fill in the fields to accomplish this task. After you have filled in the fields, press PF12 to submit your request.

```
                              Novice User Schedule                    VM:Schedule
--------------------------------------------------------------------------------

    Request name: BOSSQRY     (Your logon password:          )
    Command to execute:  QUERY1 _____________________________________________
                         _____________________________________________________

                                FIRST RUN OPTIONS

    START at: 21:00:00 (hh:mm:ss)              From: 06/20/YY (mm/dd/yy)

                                 REPEAT OPTIONS

   Run every          Hour, Day, Business Day, Week, Month,
                      Quarter    (H/D/B/W/M/Q): D

                                FINAL RUN OPTIONS

  Run the request until: __/__/__ (mm/dd/yy date of last run)
--------------------------------------------------------------------------------
PF: 1  Help     2  ...      3  End       4  Return    5  Copy      6  ...
PF: 7  ...      8  ...      9  ...      10  Print     11  ...      12  Submit
0185I ENTER DATA AND PRESS 'ENTER'.
===>

--------------------------------------------------------------------------------
```

Repeating a Request More Than Once a Day

You can repeat a request every 15 minutes, every hour, or every 3 hours. Run it all day or just between noon and 4:00 p.m. To run a request several times a day every day or on selected days of the week, combine this option with the other repeat options. Use this feature to test application programs.

Line-Mode Examples

You run a help service for your data center. Users send CMS notes containing their questions to your reader. You read the notes and send the users answers. But this afternoon you will be out of the office and Johann will take over for you. You have an EXEC called TRANSHLP that checks your reader and sends any notes to user ID JOHANN. Now you need to schedule the EXEC. You name the job TRANSJ, and run it in an EXEC called TRANSHLP JOHANN.

- To check your reader every 15 minutes between 1:00 p.m. and 6:00 p.m., enter the following command:

  ```
  vmsched schedule transj transhlp johann (at 13:00:00 to 18:00:00
  every 00:15:00
  ```

- To check your reader every hour between 1:00 p.m. and 6:00 p.m., enter the following command:

  ```
  vmsched schedule transj transhlp johann (at 13:00:00 to 18:00:00
  again hourly
  ```

- To check every hour and a quarter between 1:00 p.m. and 6:00 p.m., enter the following command:

  ```
  vmsched schedule transj transhlp johann (at 13:00:00 to 18:00:00
  every 01:15:00
  ```

- You need to run the UTILREPT job, which produces a system utilization report, every 15 minutes during your system's off peak hours, 5 p.m. to 6 a.m. To schedule this job, enter the following command:

  ```
  vmsched schedule utilrept util (every 00:15:00 fromtime 17:00:00
  totime 06:00:00
  ```

Repeating a Request Every or Every Other Day

You can process a request at the same time every day, repeat it indefinitely, or repeat it until a certain date. You can complete the following tasks with this feature:

- Check and print your daily things-to-do list
- Update databases
- Run daily system backups.

Line-Mode Examples

You keep your things-to-do list in a CMS file. Every morning when you arrive at work, you run an EXEC called TODO that prints the file if you have updated it. You named the job TODOLIST.

- To schedule the job TODOLIST to run at 6:00 a.m. so your new to-do list is ready when you arrive at work, enter the following command:

  ```
  vmsched schedule todolist todo (at 06:00:00 again daily
  ```

- To run it every business day, enter the following command:

  ```
  vmsched schedule todolist todo (at 06:00:00 again bdaily
  ```

- If you were using your to-do list only for a special project in June and July instead of permanently, enter the following command:

  ```
  vmsched schedule todolist todo (at 06:00:00 again daily from
  06/01/yy until 07/31/yy
  ```

- To run it every other day during June and July, enter the following command:

  ```
  vmsched schedule todolist todo (at 06:00:00 again bi daily from
  06/01/yy until 07/31/yy
  ```

Full-Screen Example

Using the same scenario as above, schedule your TODO EXEC to run in the morning and print your todo list during June and July every business day before you get to work. The following graphic shows how to fill in the fields to accomplish this task. Once you have filled in the fields, press PF12 to submit your request.

```
                            Novice User Schedule                    VM:Schedule
--------------------------------------------------------------------------------

    Request name: TODO____     (Your logon password:          )
    Command to execute:  TODO ________________________________________________
                         ________________________________________________

                               FIRST RUN OPTIONS

    START at: 06:00:00 (hh:mm:ss)              From: 06/01/yy (mm/dd/yy)

                               REPEAT OPTIONS

   Run every          Hour, Day, Business Day, Week, Month,
                      Quarter    (H/D/B/W/M/Q): B

                               FINAL RUN OPTIONS
  Run the request until: 07/31/yy (mm/dd/yy date of last run)
--------------------------------------------------------------------------------
PF: 1  Help     2  ...      3  End       4  Return    5  Copy      6  ...
PF: 7  ...      8  ...      9  ...      10  Print    11  ...      12  Submit
0185I ENTER DATA AND PRESS 'ENTER'.
===>
--------------------------------------------------------------------------------
```

Repeating a Request Weekly, Monthly, Quarterly, or Yearly

You can run a request once every week, month, quarter, or year. You can also schedule requests to run at the end of every month or quarter without having to specify particular dates; the server keeps track of the dates for you. You can complete the following tasks with this feature:

- Print weekly status reports
- Run monthly accounting programs and quarterly financial reports
- Send your staff reminder messages about monthly planning meetings.

Line-Mode Examples

You work in accounting. One of your responsibilities is to run a report detailing the company's use of temporary help. You have an EXEC called TEMPO that extracts data from a timesheet database and processes it. You named the job you want to run THREPORT. Enter one of the following commands, depending on how often you need to run the report:

- To run the report daily at 7:00 p.m. during the second quarter only, enter the following command:

  ```
  vmsched schedule threport tempo (at 19:00:00 again daily during 2q
  ```

- To run the report every week, enter the following command:

  ```
  vmsched schedule threport tempo (at 19:00:00 on fri again weekly
  ```

- To run the report every other Friday, enter the following command:

  ```
  vmsched schedule threport tempo (at 19:00:00 from fri again bi
  weekly
  ```

- To run the report every weekend, enter the following command:

  ```
  vmsched schedule threport tempo (at 19:00:00 on fri again weekend
  ```

- To run the report on the first of every month, enter the following command:

  ```
  vmsched schedule threport tempo (at 19:00:00 from 01/01/yy again
  monthly
  ```

- To run the report at the end of every month, enter the following command:

  ```
  vmsched schedule threport tempo (at 19:00:00 from 01/31/yy again
  monthend
  ```

- To run the report every quarter, enter the following command:

  ```
  vmsched schedule threport tempo (at 19:00:00 from 01/01/yy again
  qtrly
  ```

- To run the report at the end of every quarter, enter the following command:

  ```
  vmsched schedule threport tempo (at 19:00:00 from 03/31/yy again
  qtrend
  ```

- To run the report every year, enter the following command:

  ```
  vmsched schedule threport tempo (at 19:00:00 from 01/01/yy again
  yearly
  ```

Repeating a Request on Business Days or Weekdays

You can define your own intervals for repeating requests. Base them on business days or regular weekdays. Schedule a request to run every Monday and Wednesday each week or every ten business days. You can complete the following tasks with this feature:

- Biweekly requests
- Nonstandard accounting periods.

Business days are usually Monday through Friday, but your office might run on a different schedule. To check which days your system calls business days, enter **vmsched config** and see the lines starting MON, TUE, and so on.

Line-Mode Examples

You work in accounting. One of your responsibilities is to run the CUSTBILL EXEC that prepares customer billings. Your company's billing cycle is 4 weeks, that is, 28 days instead of a full month. You name the job you want to run CUSTBILL.

- To schedule CUSTBILL EXEC to run every 28 days at 5:00 p.m., enter the following command:

  `vmsched schedule custbill custbill (at 05:00:00 from 01/01/yy again 28 days`

- To run the CUSTBILL EXEC at a different interval, enter one of the following commands instead:
  - Every business day:

    `vmsched schedule custbill custbill (at 05:00:00 from 01/01/yy again bdaily`

  - Every 10 business days:

    `vmsched schedule custbill custbill (at 05:00:00 from 01/01/yy again 10 bdays`

  - Every Monday, Wednesday, and Friday:

    `vmsched schedule custbill custbill (at 05:00:00 from 01/01/yy on mon on wed on fri again weekly`

  - Every day of the week but Sunday:

    `vmsched schedule custbill custbill (at 05:00:00 from 01/01/yy on mon-sat again weekly`

Repeating a Request Using Time and Date Ranges

You can schedule a request to run inside or outside a general time period like weekends, holidays, day, or night. These time periods are called shifts and ranges. Shifts define time periods, such as daily work shifts. Ranges usually define date ranges, but they can define time periods too. Your system administrator sets up shifts and ranges. To find out which shifts and ranges you can use, enter **vmsched config** and see the lines starting SHIFT and RANGE.

You can complete the following tasks with this feature

- Keep daily administrative requests, like your things-to-do list EXEC, from running on weekends and holidays
- Make sure that work that uses many system resources, such as database reports, runs at night.

Line-Mode Examples

You work on a project to track the resources used by user IDs set up for temporary employees. You have an EXEC called TEMPUSE to check this. Because your department is charged for the system resources it uses, you do not want to run the report when no one is at work. Your office is open on weekends, but on a few days of the year, like Thanksgiving and Christmas, it shuts down.

- To check your system's name for these days, enter the following command:

  ```
  vmsched config
  ```

  CA VM:Schedule displays the contents of the VMSCHED CONFIG file. The lines you need to check look like this:

  ```
  RANGE 'HOLIDAY' IS BETWEEN 11/26/yy 00:00:00 AND 11/26/yy 23:59:59
  RANGE 'HOLIDAY' IS BETWEEN 11/27/yy 00:00:00 AND 11/27/yy 23:59:59
  RANGE 'HOLIDAY' IS BETWEEN 12/25/yy 00:00:00 AND 12/25/yy 23:59:5
  ```

- To schedule the TEMPUSE EXEC to run on all days except those in the range named HOLIDAY, enter the following command:

  ```
  vmsched schedule tempuse tempuse (at 08:00:00 every 04:00:00 to
  20:00:00 again daily outside holiday
  ```

You have another EXEC, THANKS, that sends a message to the skeleton staff that volunteers to work on holidays.

To schedule it to run only on holidays, enter the following command:

```
vmsched schedule thanks thanks (at 10:00:00 again daily inside holiday
```

You need to send hourly reminders to your third shift operations staff to check spool space. The THIRD shift is defined as 11 p.m. to 6 a.m. every day.

To schedule a request to send the reminders, enter the following command:

```
vmsched schedule remind remind (again hourly inside third
```

Note: If you specify an OUTSIDE shift or range without using an AT time or FROM date, *CA VM:Schedule* uses as the default the first available time and day not included in the specified shift or range. If you specify an INSIDE shift or range without using an AT time or FROM date, *CA VM:Schedule* uses as the default the first available time and day within the specified shift or range.

Chapter 4: Advanced Scheduling Techniques

This chapter describes advanced techniques you can use once you have mastered basic scheduling.

This section contains the following topics:

Retrying Requests That Cannot Run on Time (WITHIN)

You can override the default time period (WITHIN time) for how long the server continues retrying a request that cannot run at the time you specified with AT or FROMTIME. You cannot change the retry time interval.

Sometimes the server cannot start a request at the scheduled time because you are logged on, the system is down, or request processing that stopped temporarily. The server keeps trying to run the request at regular intervals for the period of time defined as the default.

To find out the default value, enter **vmsched config** and see the line starting DEFAULT LATE INITIATION TIME LIMIT.

You can complete the following tasks with this feature:

- Keep the server from continuing to interrupt your work to ask you to log off so a noncritical request can run
- Make sure that an important report processes as soon as possible after its regular start time, if the system was down at that time.

Example

You have an EXEC called CREDITS that runs monthly accounting reports. You name the job you want to run MONTHLY. When you schedule it, you want to make sure it runs even if the system goes down. Your system's default time period for retrying a request is one hour. For extra security, schedule CREDITS with a retry time of four hours instead:

```
vmsched schedule monthly credits (at 20:00:00 within 04:00:00 from
01/31/yy again monthend
```

Note: The longest time you can specify WITHIN is 99 hours, 59 minutes, and 59 seconds (99:59:59). If you specify WITHIN *, the within time is unlimited.

Running Requests That Depend on Other Requests (RELEASE)

You can run a request only after another request completes, or a file is updated, instead of running the request at a set time. You can write and then schedule a master EXEC that sets up other EXECs to process in a specific order.

You can ensure the following information with this feature:

- Data has been updated before you run reports from it
- Database updates occur in the proper order
- The database is updated before backups
- A sequence of events occurs in order, regardless of the length of each step.

Examples

You keep your department's master schedule up to date. Each employee has a data file for his or her own schedule. When someone's schedule changes, that person notifies you, and you run an EXEC called NEWSCHED to create a new master schedule. You can automate this procedure by writing an EXEC called CHKDATA to check the data files and then run NEWSCHED if any employees changed their data files since the last check. Set up a job called NEWSCHED to run as a dependent or release request:

```
vmsched schedule newsched newsched (release louisa
```

This command schedules a request called NEWSCHED to run on your user ID (GREGK) when released by user ID LOUISA. The first **newsched** is the name you give the request; the second is the command line for your NEWSCHED EXEC.

Have the CHKDATA EXEC execute a command if any employees updated their files. This command runs the request NEWSCHED that you scheduled to be released by user ID LOUISA on user ID GREGK. This command must be entered by the user ID specified after the release option on the SCHEDULE command above (LOUISA):

```
vmsched release gregk newsched
```

Finally, schedule a request called CHKDATA to run the CHKDATA EXEC at 6:00 a.m. every day:

```
vmsched schedule chkdata chkdata (at 06:00:00 again daily
```

Running Requests in Classes (CLASS)

You can run a request in a request class like those used in batch processing, instead of at a set time. Classes group requests with similar characteristics. The requests run when an operator starts the class. For example, class D might be for large requests that require a great deal of processing time. The operator might start class D after 6:00 p.m. on Monday through Thursday, and after 4:00 p.m. on Friday. The requests in the class run in the order they were scheduled.

To find out the classes in your system, enter **vmsched config** and look for the lines starting THE CLASS.

You can complete the following tasks with this feature:

- Run all your requests, if required by your system
- Use special or single-thread resources, like a plotter.

Example

On your system, class D is used for big reports that take a lot of system resources. To run your monthly accounting report (job MONACCT) with the ACCTREP3 EXEC in class D, enter the following command:

```
vmsched schedule monacct acctrep3 (class d again monthly from 01/01/yy
```

Running Requests With Specific Resource Limits (SIO)

Set the thresholds for the CPU time, SIOs, and UIOs your request uses. You can set these thresholds only if your system uses the MONITOR user exit to track the resource use. If your request exceeds the resource thresholds you set, the MONITOR user exit is invoked. What happens then depends on your system. To find out, ask your system administrator. You can complete the following tasks with this feature:

- Prevent the requests from using unexpected and excessive amounts of resources
- Test new programs.

To find out how many resources a particular request used, use the QUERY command with the LONG option.

Example

You have written a new program, REPORTER. You need to test the program and attend a meeting at 3:30 p.m. You can run the program while you are at your meeting and prevent the new program from going into an infinite loop while you are gone by limiting the amount of CPU time the request can use. The MONITOR user exit will detect excessive CPU time usage. Enter the following command:

```
vmsched schedule progtest reporter (at 15:30:00 cpu 60
```

This command specifies a threshold of 60 CPU seconds (virtual and overhead) your request can use.

You can also set the following thresholds for your request:

- *sio number* sets the threshold for the number of disk and tape I/Os your request can perform; if this threshold is exceeded, *CA VM:Schedule* calls the MONITOR user exit
- *uio number* sets the threshold for the number of unit record (reader, printer, punch) I/Os your request can perform; if this threshold is exceeded, *CA VM:Schedule* calls the MONITOR user exit

Changing Your Storage Size (STORAGE)

You can change your user ID logon storage size to run a request with different requirements from your usual ones up to your logon storage maximum.

To avoid creating performance problems during the day, use this feature to requests that require much virtual storage at night.

The storage size that you specify must be within the maximum storage size, as set for your user ID in your CP directory entry. The server does not run a request for which you give an invalid storage size.

Note: You can use *CA VM:Secure* or *CA VM:Director* to find out the maximum storage size for your user ID. Use selection 1 on the *CA VM:Secure* or *CA VM:Director* User Selection Menu to review your storage size and maximum size.

Example

In this example, the logon storage size of your user ID is 3M. You have a maximum storage of 8M, and you want to process a large document, which requires 5M of storage. To schedule the work as request PAPERS, and request 5M of space to run it in for 7:00 tonight, enter the following command:

```
vmsched schedule papers (at 19:00:00 storage 5m
```

Because you did not enter a command name, the server responds with the following prompt:

```
ENTER COMMAND PARAMETERS:
```

Now, enter the command line:

```
reformat survey script c (continue page (from 26 to 318)
```

The server prompts you for the command-line parameters because it contains parentheses. *CA VM:Schedule* cannot distinguish between your parentheses and its own. If your command contains a parenthesis, you must either leave it out and let the server prompt you for it, or put it in an EXEC and then schedule that EXEC instead.

Controlling the Request Log File (LOG)

You can change how often the server sends you a log file about what happened when it ran your request. This change requires LOG authorization. Normally, you receive a request log file when a request does not run, when it stops because of an exceptional event such as an error, or when it runs for the last time.

You can complete the following tasks with this feature:

- Receive positive verification each time an important request runs
- Have the log file of an important request sent to another user
- Avoid creating log files for requests that seldom fail or that you check on normally.

You can use the QUERY command to review the same information that appears in a request log, along with other request status information. For more details, see *Listing Request Information (see page 48).*

If you schedule a request using the request execution monitor, a *CA VM:Schedule* feature your system may use for request tracking and control, the server sends you a second log file containing more extensive reporting about how the request ended. See *Monitoring Requests* (see page 45) for more information about the request execution monitor.

Examples

You have an EXEC called DAYREPT that you want to run each day when you finish work at 5:00 p.m. To receive a log file each time your request runs, enter the following command:

```
vmsched schedule dayrept dayrept (at 17:00:00 again daily log
```

To have your associate, Alice, receive the log file, enter the following command:

```
vmsched schedule dayrept dayrept (at 17:00:00 again daily log spool
alice
```

To receive a log file only when the request does not complete successfully (for instance, it skips a request or produces execution errors), enter the following command:

```
vmsched schedule dayrept dayrept (at 17:00:00 again daily except
```

To never receive a log file, error messages, or the notice of the request's last run, enter the following command:

```
vmsched schedule dayrept dayrept (at 17:00:00 again daily nolog
```

Running EXECs on CA VM:Schedule (EXEC)

You can run EXECs on the *CA VM:Schedule* service virtual machine instead of your user ID, so you do not need to log off when they run. Your system administrator must give you authorization to run EXECs.

The server provides two EXECs that run on the *CA VM:Schedule* service virtual machine: VMDDOS and VMDMSG. VMDDOS sends DOS/VSE requests from your user ID to the reader of a DOS/VSE guest virtual machine. VMDMSG lets you send messages to users, including yourself, at predetermined times. VMDMSG is especially useful for reminders.

Your data center might restrict use of these EXECs, and it might have other EXECs you can use. To find out which EXECs you are authorized to use, enter **vmsched config exec**.

The server displays a list like the following example:

```
USER IS AUTHORIZED FOR VMDMSG EXEC
USER IS AUTHORIZED FOR VMDDOS EXEC
```

Scheduling EXECs to run on the server similar to scheduling EXECs to run on your user ID. The major difference is that the command you use is EXEC, not SCHEDULE. Also, you cannot run these EXECs in request classes, specify resource limits, or change the logon storage size.

Examples

Your boss is very busy this week. Remind your boss that you are having lunch together today:

```
vmsched exec lunch vmdmsg theboss remember lunch at 12 (at 11:15:00
```

You are going on vacation for a few weeks. Make sure your co-worker, Frank, remembers to water your plants and check your interoffice mail for responses to the Help Wanted advertisement you are running in the paper:

```
vmsched exec remind1 vmdmsg frank please water plants today (at
15:00:00 on fri again weekly from 02/10/yy until 02/28/yy
```

```
vmsched exec remind2 vmdmsg frank check mail for good resumes (at
13:00:00 from 02/10/yy until 02/28/yy again daily
```

Restrictions

Using the EXEC command, you can execute only EXECs on the *CA VM:Schedule* service virtual machine. The EXECs run under CMS subset and should be very short, because the server cannot process any other work while the EXECs run.

You cannot run EXECs that modify user storage on the CA VM:Schedule service virtual machine. If an EXEC contains *CA VM:Schedule* commands, the commands must be preceded by CP SMSG *, not VMSCHED. For example, to schedule an EXEC called WEEKRPT at 11:00, enter the following command:

```
cp smsg * exec weekrpt weekrpt (at 11:00:00
```

When an EXEC runs on an operational *CA VM:Schedule* system, the server stacks the user ID that scheduled the EXEC and then invokes the EXEC with its parameters. If an EXEC running on the *CA VM:Schedule* service virtual machine must use the stack, a read of the first stacked line or a DESBUF must be performed at the start of the EXEC.

Important! Do not use the CMS WAKEUP command in EXECs that run on the *CA VM:Schedule* service virtual machine. Using WAKEUP can impact the ability of the product to operate and run scheduled requests. Any program that intercepts timer interrupts has the potential of impacting *CA VM:Schedule's* ability to operate correctly.

To find out more about EXECs that run on the *CA VM:Schedule* service virtual machine, contact your system administrator.

VMDMSG Return Codes

The VMDMSG EXEC returns a non-zero return code in the following situations:

Return Code and Meaning

20

You did not specify a user ID.

45

The user ID you tried to send a message to is not logged on or is invalid.

57

The user ID you tried to send a message to is disconnected or has MSG set off.

Monitoring Requests (MONITOR)

You can run requests under control of the *CA VM:Schedule* request execution monitor to produce status information not available directly under CMS. This information is useful when determining why a request did not complete successfully. The request execution monitor lets you complete the following tasks:

- Send a request completion time and command return code to the *CA VM:Schedule* service virtual machine and a specified user ID

  In addition, the monitor sends the elapsed time to the *CA VM:Schedule* service virtual machine.
- Spool your console to your own or another user ID
- Request a CP or VMDUMP of the virtual machine on which the request runs if the request abends
- Log off your user ID after the request has completed.

Example

Schedule the DAYREPT request to run every day at 8 p.m. Your user ID is to be logged off when the request has completed. Spool the console to user ID ALEX. If your user ID abnormally terminates while running the request, *CA VM:Schedule* is to perform a VMDUMP of your user ID and send the dump to user ID ALEX:

```
vmsched schedule dayrept dayrept (again daily at 20:00:00 dump vm
dumpto alex consto alex logoff yes
```

Note: If you monitor requests, make sure your user ID has access to the *CA VM:Schedule* communications program, VMSCHED MODULE. If *CA VM:Schedule* autologs a user ID and cannot find this module, the request does not run and the user ID sits idle for 15 minutes until it is forced off by the system. (After the user ID is forced, the QUERY command shows the status UNKNOWN TERMINATION.)

Chapter 5: Reviewing Request Times/Status and System Defaults

You can use the WHEN, QUERY, and CONFIG commands to find out information about your requests and how your system defaults are set up.

This section contains the following topics:

Listing Run Time (WHEN)

To find out when a scheduled request runs, use the WHEN command. Check the time of the next run, several of the next runs, or all runs within a given date range.

You can complete the following tasks with this feature:

- Make sure that you scheduled a request the way you intended
- Check if any requests are scheduled to run when the system shuts down, and take action to change the run times.

Examples

You just scheduled a request named CREDITS. You used a specification you have not used before, MONTHEND. The request should run at the end of every month, but you want to make sure. To check the day, date, and time of the request's future runs, up to 12/31, enter the following command:

```
vmsched when credits (until 12/31/yy
```

You scheduled another request, DEBITS, to run for a few months, but you cannot remember if you had it stop running in March or April. To find out, enter the following command:

```
vmsched when debits (from 03/01/YY until 04/30/yy
```

CA VM:Schedule displays the time and date of all request runs within the date range March 1 through April 30.

You scheduled another request, CHECKUP, to run every 15 minutes today. To make sure the request was scheduled correctly, display the next 10 runs:

```
vmsched when checkup (next 10
```

CA VM:Schedule displays the time of just the next 10 runs for today only. The runs are consecutive, starting with the upcoming one. If you leave out **next**, **from**, and **until**, *CA VM:Schedule* lists the times and dates for the next 999 runs today. If you specify just dates, *CA VM:Schedule* lists up to 999 runs for the date range specified.

Listing Request Information (QUERY)

You can review a list and brief description of all your scheduled requests. The list includes all requests that still have runs left. The list can also include requests that you canceled recently.

You can complete the following tasks with this feature:

- Find the name of a request you want to change or keep from running
- Determine what happened the last time that a request ran
- Determine the resources that the request consumed
- Check how you scheduled a request, so you can schedule another one the same way.

Examples

You want to cancel a request you scheduled, but you cannot remember the name you gave the request. To get a list of all the requests for your user ID, enter the following command:

```
vmsched query *
```

The server displays a list giving the names of all your requests, their execution status (what happened last time they ran), the time and date of their next and last runs, and the user ID on which they run.

To get a brief report on just one request, the one named BOSSREPT, enter the following command:

```
vmsched query bossrept
```

The server displays the same information for just that request.

Output Definitions

REQUEST

Specifies the name you gave the request when you scheduled it.

EXECUTION STATUS

Specifies what happened the last time the request ran. The following table presents possible statuses:

Status	Meaning
BATCH JOB READIED	*CA VM:Schedule* signaled *CA VM:Batch* to run a *CA VM:Batch* request.
BATCH RUN PENDING	*CA VM:Schedule* signaled *CA VM:Batch* to start a request but a run of that request is currently in progress. The next run of the request is pending the current run's completion.
CANCELED- AUTOLOG ERROR	Canceled due to an XAUTOLOG command error.
CANCELED- ON HOLD	Canceled and request is on hold.
DATE ADJUSTED	*CA VM:Schedule* changed the original run date to meet your other specifications, like MONTHEND.
EXECUTION COMPLETE	Ran as scheduled.
EXECUTION ERROR	Could not run. For example, if your request were to send a message to someone who was not logged on.

Status	Meaning
INITIATION DELAYED	Run time changed with DELAY FOR or DELAY UNTIL.
INITIATION FAILED	Could not start, probably because the user ID it was to run on was disconnected or logged on.
INITIATION SKIPPED	Not run because you used the SKIP command or the system was down at the time, or because the request did not run before the WITHIN time was exceeded.
JOB NOT FOUND	Could not run because the *CA VM:Batch* job associated with this schedule request no longer exists.
JOB NOT RUN	Could not run because the *CA VM:Batch* job was not associated with a *CA VM:Schedule* request; the *CA VM:Batch* job has no scheduling options.
NORMAL INITIATION	Started without error.
NOT PREVIOUSLY RUN	Not executed yet.
NOT RESCHEDULED	Not scheduled to run again.
PURGE SCHEDULED	Canceled or finished running for its last scheduled time, so all information about it will be erased from the *CA VM:Schedule* database.
READY CMD ERROR	Could not run the scheduled *CA VM:Batch* job because of an error in the *CA VM:Batch* READY command.
REQUEST CANCELED	Canceled and will not run again.
SKIPPED TO COMPLETION	Kept from running with SKIP so many times or for so long a period that there are no runs left.
UNKNOWN TERMINATION	Started normally, but *CA VM:Schedule* does not know how it completed. The user ID may have gone into a VM or CP READ, or been forced off by the system. This status can also occur when a request is to be monitored using the job execution monitor, but the autologged user ID does not have access to the *CA VM:Schedule* communications modules (VMDCOM and VMSCHED).
USER HOLD REMOVED	Was kept from running indefinitely. HOLD has since been removed, but request has not run yet.
USER REQUEST ABENDED	Started but could not complete because of a serious error.
USER REQUEST ON HOLD	Kept from running indefinitely, either by HOLD ON or because *CA VM:Schedule* encountered an error.
USER WAS LOGGED ON	Could not run because the user ID on which it was to run was logged on.

Status	Meaning
CA VM:Batch NOT AVAIL	Could not run a *CA VM:Batch* request because *CA VM:Batch* was not running.
WITHIN TIME EXPIRED	Could not run because the time specified with WITHIN has passed.

LAST DATE/TIME

Specifies when the request ran last. If the time is followed by an asterisk, it means the last run of this scheduled request was initiated with the RELEASE command.

NEXT DATE/TIME

Specifies when the request will run next.

USERID

Specifies the user who owns the job.

Reviewing Detailed Status Reports (QUERY)

Review a detailed status report about a request you scheduled or for all your scheduled requests. The report includes all your specifications for when and how the request should run, and the command-line *CA VM:Schedule* processes.

You can complete the following tasks with this feature:

- Determine what occurred when the request ran last
- Review the specifications for a particular request
- Find out the next run time for the request
- Determine how many resources the request consume.

Example

You have a request called REPORTX. You want to schedule another request called REPORTY in the same way, but you do not remember just how you scheduled REPORTX. To find out, enter the following command:

```
vmsched query reportx (long
```

The server displays a screen of information including the EXEC or command you scheduled, your scheduling specifications, the request's last and next run times, and what happened the last time it ran.

Output Definitions

Most fields in the QUERY command output have the same names as the options in the SCHEDULE Command (see page 112) or EXEC (see page 43) command. The other items are:

COMMAND

Specifies the command line for an EXEC or request scheduled with the EXEC or SCHEDULE command.

END TIME

Specifies when the request actually finished the last time it ran.

JOBID

Specifies the jobid of the CA VM:Batch job associated with this request.

LAST-RUN

Specifies when the request ran last. If the time is followed by an asterisk, it means the last run of this scheduled request was initiated with the RELEASE command.

LOGOFF

Tells whether the user ID the request runs on is automatically logged off when the request finishes processing.

NEXT-RUN

Specifies when the request will run next.

RC

Specifies the return code with which the request ended.

REQUEST MONITOR

Tells whether your request ran using the *CA VM:Schedule* request execution monitor. You can choose whether to use request monitoring for each request you schedule. If you do not specify monitoring options, *CA VM:Schedule* uses the system default for your requests. To find out what the default is, enter **vmsched config** and look for the line starting USERS.

SEND CONSOLE TO

Specifies the user ID to which the console listing is sent.

SEND DUMP TO

Specifies the user ID to which a dump is sent.

SPOOL CONSOLE

Specifies whether your console is spooled while the request runs.

STATUS

Specifies what happened the last time the request ran. For descriptions of these entries, see *Listing Request Information* (see page 48).

SYS-EXEC

Specifies the command line for an EXEC scheduled with EXEC.

RESOURCE LIMITS

Specifies the maximum CPU time, SIOs, or UIOs the request can use.

RESOURCE USAGE

Specifies the CPU time, SIOs, or UIOs consumed by the last run of the request.

ELAPSED TIME

Specifies the time the request took to run.

Sending List or Status Output to File, Printer, or Stack (QUERY)

You can have the output for the request lists and status reports (created by the WHEN or QUERY command) put in a file, sent to your virtual printer, or can put in the program stack, instead of displayed on your terminal. You can have the output sent to more than one location at a time, or to the *CA VM:Schedule* service virtual machine console.

You can complete the following tasks with this feature:

- Get printed status reports for your requests
- Keep a copy of scheduling specifications of a request for your future reference
- Have your EXEC read the status output in the program stack, and you take action from this information.

Examples

To get a list and brief status report of all your requests and put it in the program stack in first-in, first-out order, enter the following command:

```
vmsched query * (stack fifo
```

To put the same list in the program stack in last-in, first-out order, enter the following command:

```
vmsched query * (stack lifo
```

To get a printed list of the times your request TODO runs in January, enter the following command:

```
vmsched when todo (from 01/01/yy until 01/31/yy print
```

To get a detailed (long) status report for the same request and have it both put in a file named TODOTIME VMSCHED A and displayed on your screen, enter the following command:

```
vmsched query todo (long file todotime term
```

Listing System Definitions and Defaults (CONFIG)

You can find out how your data center has set up *CA VM:Schedule*. You can display the system defaults on your terminal. The display includes what shifts, ranges, and classes in which you can run requests. The display also includes which days of the week are business days, and when quarters of the year start and end.

You can complete the following tasks with this feature:

- Find out valid classes and resource limits
- Check the name and time of a shift or range that you want to use
- Avoid entering unneeded items like WITHIN in your SCHEDULE command.

To list system defaults, enter the following command:

```
vmsched config
```

Output Definitions

The server displays the following types of information on your terminal:

AUTOLOG RETRY INTERVAL IS 30 SECONDS.

If you are logged on when a scheduled request runs on your user ID, the server checks every 30 seconds to see if you logged off. If you are still logged on, the server sends you a message.

PURGING IS DONE 5 DAYS AFTER REQUEST COMPLETION.

After you cancel a request, or it runs for the last time, information about the request remains in the server database for five days. Until then, you can use QUERY to check the status of the request, and you can update the request.

DEFAULT LATE INITIATION TIME LIMIT IS 00:02:00.

If a request cannot run at its scheduled time because you are logged on or the system is down, the server keeps trying to run it for 2 minutes after the original time. Then it skips that run of the request. You can override this behavior by using WITHIN when you schedule the request with SCHEDULE or EXEC.

DEFAULT INITIATION TIME IS 00:00:05 AFTER SCHEDULING.

If you do not use AT or FROMTIME in your SCHEDULE or EXEC command, the request first runs 5 seconds after you enter the command. The delay time applies only to requests scheduled to run today. For example, at 11:00 today you schedule a request to run tomorrow without specifying the start time, the server schedules the request to run tomorrow at 11:00:00, instead of 11:00:05.

SHIFT 'FIRST' IS BETWEEN 06:00:00 AND 16:00:00.

Your site has a shift named FIRST that runs from 6 a.m. to 4 p.m. every day.

RANGE 'HOLIDAY' IS BETWEEN 01/01/02 00:00:00 AND 01/02/02 23:59:59.

RANGE 'HOLIDAY' IS BETWEEN 11/26/02 00:00:00 AND 11/27/02 23:59:59.

RANGE 'HOLIDAY' IS BETWEEN 11/27/02 00:00:00 AND 11/28/02 23:59:59.

RANGE 'HOLIDAY' IS BETWEEN 12/25/02 00:00:00 AND 12/26/02 23:59:59.

Your system has a range named HOLIDAY that includes New Year's Day, Thanksgiving, the day after Thanksgiving, and Christmas.

MON IS DEFINED AS A BUSINESS DAY.

TUE IS DEFINED AS A BUSINESS DAY.

WED IS DEFINED AS A BUSINESS DAY.

THU IS DEFINED AS A BUSINESS DAY.

FRI IS DEFINED AS A BUSINESS DAY.

SAT IS DEFINED AS A WEEKEND DAY.

SUN IS DEFINED AS A WEEKEND DAY.

Your system considers Monday through Friday as business days. Saturday and Sunday are weekend days.

THE RANGE FOR 'QUARTER1' IS BETWEEN 01/01/*yy* AND 03/31/*yy*.

THE RANGE FOR 'QUARTER2' IS BETWEEN 04/01/*yy* AND 06/30/*yy*.

THE RANGE FOR 'QUARTER3' IS BETWEEN 07/01/*yy* AND 09/30/*yy*.

THE RANGE FOR 'QUARTER4' IS BETWEEN 10/01/02 AND 12/31/02.

The quarters at your site are the traditional ones that start at the beginning of January, April, July, and October.

THE CLASS 'A' AUTOLOG LIMIT IS 3; WARNINGS WILL NOT BE SENT.

RESOURCE LIMITS ARE CPU 0, SIO 0, AND UIO 0.

MAXIMUM AUTOLOG RETRY ATTEMPTS IS 2 PER REQUEST.

Only three requests can run in class A at a time. After that, other requests wait until one original request finishes. The class has no resources limits.

The server does not try to autolog users who log on when a scheduled request runs. Therefore, CLASS requests for logged-on users requeue immediately. In this case, the servers attempts to start the request two times. After the attempts, the server skips the request. The server does not warn users that it is trying to autolog their user IDs.

CA VM:Schedule IS INITIATING ALL REQUESTS.

CA VM:Schedule is processing requests normally.

USERS MAY CHOOSE TO USE THE REQUEST EXECUTION MONITOR.

THE DEFAULT IS NO.

Users can choose whether to use the *CA VM:Schedule* request execution monitor. By default, the system does not use the request monitor.

Chapter 6: Manipulating Scheduled Requests

Once you schedule a request, you can complete the following tasks:

- Put the request on hold and then restart it
- Change the parameters
- Skip a run
- Cancel the request.

This section contains the following topics:

Changing the Run Time (DELAY)

You can change the time that a request is scheduled to process for just the next run or for all the runs it has left.

You can complete the following tasks with this feature:

- Keep a request from interrupting your work
- Adjust a request when your schedule or the system schedule changes

Examples

You have a request named TODO. Every morning at 7:00, it runs the TODO EXEC to check your things-to-do file, and prints a copy for you, if you changed it. For example, you arrived early this morning. You are hard at work and do not want to log off just so your to-do list can print. To make the request run an hour later only this one time, enter the following command:

```
vmsched delay todo for 01:00:00
```

This delay does not affect subsequent runs of the request.

If you started work earlier every day, you could permanently change the run time to 6:00 a.m. by entering the following command:

```
vmsched delay todo until 06:00:00
```

You can schedule a request to run earlier than originally scheduled by specifying an UNTIL time between the present time and the next scheduled run time of the request. Further, you can schedule a request to run later than originally scheduled by specifying an UNTIL time later than the originally scheduled time. Any subsequent runs of the request are based on the time that you specify with the DELAY command.

Skipping One or More Runs (SKIP)

You can skip one or more subsequent runs of a request without canceling the whole request. The runs must be consecutive; for example, you cannot use a single command to skip the first run, run the second, and skip the third.

You can complete the following tasks with this feature:

- Keep administrative requests such as status reports from running while you are on vacation.
- Cancel some runs of a report you need most of the time. Schedule the report to run every day, then skip the runs that you do not need to save system resources.

Example

You have a request named BOSSREP that sends a copy of your status report to your manager every week. Your manager is away for the next three weeks, and prefers to receive a combined report on returning to work. To cancel the next three runs of your request, enter the following command:

```
vmsched skip bossrep 3
```

Restrictions

You cannot skip runs for a request that are scheduled to be released. Also, you can only skip requests that you scheduled to repeat. For a request with only a single run, delay or cancel the request to avoid running it. For more details, see *Canceling a Request* (see page 62) and *Changing the Run Time* (see page 59).

Putting Requests on Hold (HOLD ON)

You can keep a scheduled request from running indefinitely. This feature lets you stop running a report that you do not need for an extended or indefinite time.

If a request is on hold when its scheduled run time occurs, the request does not run. However, if the request is taken off hold before your system WITHIN time passes, the server tries to start the request during the WITHIN period.

Note: The run times that are missed are skipped, not adjusted, so be careful when putting requests with a definite stop date on hold. If you do not start the request again before the stop date, it never runs again.

Example

You have a request named REPORTX that runs a report for a special project. The project has been suspended. The project may reactivate in the future, but you do not know when. You do not want to cancel the request altogether, but you do want to keep it from running indefinitely. Enter the following command:

```
vmsched hold on reportx
```

Restrictions

Consider the following restrictions when you put requests on hold:

- You cannot hold a request that is attempting to run.
- You cannot skip or delay a request on hold.
- Copies of requests on hold are treated as new requests and are not put on hold.

Restarting a Request (HOLD OFF)

You can restart a request that either you or *CA VM:Schedule* put on hold. *CA VM:Schedule* may stop running your request and put it on hold because of an error; when this happens, you get a message indicating the error. You must restart the request if you want it to run again. If you are not sure which of your requests have been put on hold, enter the following command:

```
vmsched query *
```

Example

You received a message file this morning which informed you that the server put your request REPORTS on hold because of an autolog error. Your system administrator informs you that the problem was solved. To restart the request to run again as scheduled, enter the following command:

```
vmsched hold off reports
```

Automatic System Hold

If one of the following events occurs, the server can place your request on hold automatically:

- An autolog error terminates the request start.
- You scheduled an EXEC, but the request cannot execute. The *CA VM:Schedule* service virtual machine did not locate the EXEC on any of its accessible disks.
- You scheduled a request using the EXEC command specifying an EXEC you are no longer authorized to use.
- The logon storage size that you specified in the storage scheduling option now exceeds the maximum size that is allowed for your user ID.
- The server encountered an error while trying to change your logon storage size to the value specified in your request.

Canceling a Request (CANCEL)

You can cancel a request and can delete it from the system before all the scheduled runs have finished. Complete this process and cancel a request you no longer need.

Example

You worked on a project for which you had to run a weekly report, REPORTX. The project completed and you no longer need the request. To cancel the report, enter the following command:

```
vmsched cancel reportx
```

Changing Scheduling Parameters (CHANGE)

You can use the CHANGE command to modify scheduled requests. You can also use CHANGE to change *CA VM:Batch* scheduled jobs through the *CA VM:Schedule* interface to *CA VM:Batch*.

Note: For more information about using the interface, see the *CA VM:Batch User and Group Manager Guide*.

Examples

You want to schedule a request named STATJOB that executes a program named STATUS. You schedule this request to run every Friday at 9:00 p.m., while you are not in the office. The following command schedules this request:

```
vmsched schedule statjob status (at 21:00:00 on fri again weekly
```

You decide that the request must run daily. To modify the request schedule, enter the following command:

```
vmsched change statjob (again daily
```

To verify your changes, enter the following command:

```
vmsched query statjob (long
```

Changing a Request in Full-Screen

The full-screen CHANGE command is available through all user level main menus. To use CHANGE from the menu:

1. Select CHANGE from your User Main Menu and press ENTER. The Change a Request screen displays and prompts you for the name of the request you want to change.
2. Enter the name of the request and press ENTER. The Request Change screen appears at your level of expertise, with the request information populated. The following graphic shows a Novice Request Change screen.

3. To make your changes, edit the appropriate fields and press PF11 (Change) to save your changes.

```
                          Novice Request Change                    VM:Schedule
--------------------------------------------------------------------------------

    Request name: REPORT      (Your logon password:          )
    Command to execute:  REPORT ACCT
                         ____________________________________________________

                               FIRST RUN OPTIONS

    START at: 07:00:00 (hh:mm:ss)              From: 03/20/yy (mm/dd/yy)

                                REPEAT OPTIONS

   Run every          Hour, Day, Business Day, Week, Month,
                      Quarter    (H/D/B/W/M/Q): D

                               FINAL RUN OPTIONS

  Run the request until: __/__/__ (mm/dd/yy date of last run)
--------------------------------------------------------------------------------
PF: 1  Help     2  ...      3  End       4  Return    5  ...       6  ...
PF: 7  ...      8  ...      9  ...      10  Print    11  Change   12  ...

===>
--------------------------------------------------------------------------------
```

Chapter 7: Running Requests on Other User IDs

You can schedule requests to run on user IDs other than your own, list information about those requests, and change them.

You can complete the following tasks with this feature:

- Work on your own user ID while running a request on another user ID.
- Autolog administrator user IDs to perform system maintenance functions such as backing up spool files or automating your system backups.
- Autolog user IDs that you are responsible for at regular intervals. Regularly autologging user IDs prevents them from being put on hold or deleted due to inactivity.

To schedule a request on another user ID, use the appropriate commands and add the word *user* and a user ID at the end of the command. The user ID is the user ID on which you want the request to run. When you use commands for EXECs scheduled to run on the server, the user ID is the user ID that owns the request.

When you enter the command, the server prompts you to enter the CP logon password for the user ID. When you enter the password, it does not appear on the screen.

The following examples illustrate some scheduling tasks that you can perform on other user IDs:

- Schedule an EXEC called REPORTX that runs every day at 5:00 a.m. on your accounting administration user ID, ACCTREP:

  ```
  vmsched schedule reportx (at 05:00:00 again daily user acctrep
  ```

- List the next six runs of REPORTX for today:

  ```
  vmsched when reportx (next 6 user acctrep
  ```

- List all the requests that run on the ACCTREP user ID; put the list in a file named ACCTJOBS VMSCHED on your A-disk:

  ```
  vmsched query * (file acctjobs user acctrep
  ```

- Change the time REPORTX runs from 5:00 a.m. to 8:00 a.m. for the next run only:

  ```
  vmsched delay reportx for 03:00:00 (user acctrep
  ```

- Change the time REPORTX runs from 5:00 a.m. to 8:00 a.m. from now on:

  ```
  vmsched delay reportx until 08:00:00 (user acctrep
  ```

- Keep REPORTX from running its next two scheduled times, without canceling the rest:

  ```
  vmsched skip reportx 2 (user acctrep
  ```

- Suspend REPORTX from running until you restart it:

  ```
  vmsched hold on reportx (user acctrep
  ```

- Restart REPORTX after you put it on hold:

  ```
  vmsched hold off reportx (user acctrep
  ```

- Cancel REPORTX so it never runs again:

  ```
  vmsched cancel reportx (user acctrep
  ```

You can change the user ID on which a request runs even after it is scheduled. You must have TRANSFER authorization to change the user ID. This feature is useful for reassigning requests that belong to an employee who has left your company or changed departments.

Consider the following example:

- ROGER has left the company. You must make his supervisor, LOUISE, the owner of all his requests. Enter the following command:

  ```
  vmsched transfer * roger louise
  ```

- ALICE has been promoted. She keeps most of her jobs, but one, ACCOUNTS, should now belong to LOUISE. Enter the following command:

  ```
  vmsched transfer accounts alice louise
  ```

The server checks for duplicate request names when you transfer requests. You cannot transfer a request to user ID if that user ID already owns a request with that name. If you try to do so, the server rejects the transfer with an error message and a CMS return code of 24.

Chapter 8: Command Reference

The following table lists and briefly describes the commands for general *CA VM:Schedule* users in alphabetical order.

Command	Description
CANCEL	Cancel all future runs of a request
CHANGE	Change request parameters
CONFIG	List *CA VM:Schedule* system defaults and definitions
DELAY	Change the time a request runs
EXEC	Schedule EXECs to run on the *CA VM:Schedule* service virtual machine
HOLD	Suspend or resume request initiations
QUERY	List the status of requests
RELEASE	Run a dependent request scheduled with the SCHEDULE command's RELEASE option or initiate a scheduled request to run now
SCHEDULE	Schedule a command, EXEC, or other program to run on your user ID or a user ID to which you have access
SET DISPLAY	Set screen expertise level
SKIP	Skip the next one or more runs of a request
TRANSFER	Change the user ID on which a request rungs
WHEN	List the times a request runs

CANCEL Command

The CANCEL command cancels any further initiations of a single request, or cancels all requests for a specified user ID.

```
CANcel {requestname | *} [([User userid] [Password password]]
```

Definitions

requestname

Specifies the name of the request you want to cancel. Once a request is canceled, you can use that request name again for a new request. Use an asterisk to cancel all requests for the specified user ID. To specify all request names that match the characters preceding the asterisk, use a trailing asterisk in the request name as a wildcard character.

USER *userid*

Specifies the user ID of the virtual machine whose requests are being canceled. If you do not specify the USER option, the default is your user ID.

PASSWORD *password*

Specifies the CP logon password of the virtual machine whose requests are being canceled. If the CANCEL command is invoked through the CP SMSG command, the password must be explicitly specified on the command line. Some sites do not allow users to specify a logon password directly on the *CA VM:Schedule* command line. If your site has this limitation, the server prompts you to enter the password after you issue the command.

Description

The CANCEL command cancels any further initiations of a single request or all requests for a specified user ID. The record of each canceled request remains in the request database for the number of days specified by your site. To find out how long canceled requests remain in the database, use the CONFIG command. Request database purging occurs at midnight or the next time the server starts.

If you run the CANCEL command on a request and later you decide to retain the request, remove the CANCEL flag from the request. To retain a request, remove the CANCEL flag before the system purges the request from the request database.

To remove the CANCEL flag from a request, do one of the following actions:

- Execute the CHANGE command against the request. Make an arbitrary change to the request.
- Change the request through the *CA VM:Schedule* screens.

Example

To cancel all further initiations of your request PROD, enter the following command:

```
vmsched cancel prod
```

Example

To cancel all requests that belong to LYDIA, enter the following command:

```
vmsched can * (user lydia
```

CHANGE Command

The CHANGE command modifies scheduled requests after you submit them.

```
CHange requestname [command] [( "Options"]

Options:
["Initial date/time options"] [Within {hh:mm:ss | *}]
["Repeat options"]
["Resource limit options"]
["Initiation limits options"]
["Shift and range options"]
["Log file options"]
["Monitoring options"]
["Miscellaneous options"]
```

```
Initial date/time options:

  Class class
| Release userid [Within {hh:mm:ss | *}]
| “Event Time Options” [Within {hh:mm:ss | *}]


Event Time Options:

[AT hh:mm:ss]
[FROMTime hh:mm:ss]
[On day]
[FRom “From options”]


From options:

  mm/dd/yy
| dayname
| [“ParmA” [“ParmB” [“ParmC”]]]


ParmA:

{nnn | F | L}


ParmB:

{dayname | WE | W | B}


ParmC:

{weeknumber | M | monthname  | Q | nQ}

Repeat options:

[AGain [BI] interval]
[ON day[-day]]...
[Every hh:mm:ss]
[DURing {nnn | M | monthname | Q | nQ}]

Resource limit options:

[CPu seconds]
[Sio count]
[Uio count]
[STorage size]

Initiation limits options:

[For number]
```

```
[Totime hh:mm:ss]
[UNtil "Until options"]


Until options:

  mm/dd/yy

| dayname
| [“ParmD” [“ParmE” [“ParmF”]]]


ParmD:

{nnn | F | L}


ParmE:

{dayname | WE | W | B}


ParmF:

{weeknumber | M | monthname | Q | nQ}


Shift and range options:

[Inside {shift | range}]
[OUtside {shift | range}]


Log file options:

[EXCept [SPool userid]]
[Log [SPool userid]]
[Nolog]


Monitoring options:

[Monitor {Yes | No}]
[LOGOff {Yes | No}]
[CONSole {Yes | No}]
[CONSTo userid]
[DUMP {CP | VM | N}]
[DUMPTo userid]


Miscellaneous options:
```

```
[Password password]

[SYSname {systemname | *}]
[USer userid]
```

Definitions

requestname

Specifies the name of the request to be changed.

command

Specifies the EXEC, program, CP command, or CMS command to be executed when the request runs. If specified on the *CA VM:Schedule* command line, the command parameters cannot contain parentheses. The maximum command parameter length is 108 characters, including filename, blanks, and punctuation.

The command line is passed to the console stack of the user ID when the program autologs the user ID.

The PROFILE EXEC of any virtual machine that the server autologs must not contain any commands that alter or destroy the line that the server places in the console input buffer (sometimes referred to as the console stack). If this console stack occurs, or the first read to the console is issued prematurely, the scheduled request cannot execute as expected, even though the product autologs the virtual machine successfully. For more information about the console stack, refer to the CMS user guide appropriate for your site.

The PROFILE EXEC of any virtual machine that CA VM:Schedule autologs must not prompt for parameters when the virtual machine is running disconnected.

Initial Date/Time Options

AT *hh*:*mm*:*ss*

Specifies the time of day to initiate the request. If you do not specify AT nor FROMTIME, the time of day defaults to a site-specified delay interval from the moment you issued the command. See *Start Times* (see page 97) for more information about when requests are scheduled to start.

CLASS *class*

Specifies the initiation class for the request. *class* is the one-character class identifier under which the command is to operate. Valid class values are A-Z and 0-9, and your system administrator defines these values. You cannot use this option with the following options:

- AGAIN HOURLY
- AT *hh:mm:ss*
- EVERY *hh:mm:ss*
- FROMTIME hh:mm:ss
- INSIDE range
- OUTSIDE range
- RELEASE userid
- TOTIME hh:mm:ss
- WITHIN hh:mm:ss

Note: Requests scheduled with the CLASS option are initiated by a *CA VM:Schedule* operator or through a schedule determined by your site. You can reschedule classed requests, but not more than once per day. To determine what classes (and resource limits per class) have been specified at your site, use the CONFIG command.

Note: For information about the request processing classes that correpond to the SCHEDULE CLASS option, see "CLASS: Defining Classes" in the *CA VM:Schedule Administration Guide*.

FROMTIME *hh*:*mm*:*ss*

Specifies the time of day at which the EXEC is to initiate. Use FROMTIME with TO to specify a time period that spans midnight. If FROMTIME is used with AT time, the FROMTIME option takes precedence. See *Start Times* (see page 97) for more information about when requests are scheduled to start.

FROM *mm/dd/yy*

Specifies the date of the first (or only) request initiation. If the FROM option is not specified, the first initiation defaults to the current date unless you specify that initiation is to occur on certain days of the week or the specified time has passed for the current day.

All dates that you specify with the FROM option must contain a month and a day. If you omit the year, the default is the current year. The latest date that you can specify is December 31, 2041. Year ranges from 00 through 41 are taken to be in the 21st century.

FROM *dayname*

Specifies the day of the first (or only) request initiation. If *dayname* is today, the server schedules the request to run a week from today. If the FROM option is not specified, the first initiation defaults to the current date unless you specify that initiation is to occur on certain days of the week or the specified time has passed for the current day.

FROM *options*

Schedules a request's first, or only initiation on a given day, or week of a particular week, month, or quarter. The following table describes the options and meanings:

Option	Meaning
nnn	Starts the request on the nth day or week. For example, to start a request on the second Monday in January, use these FROM options: `from 2 mon jan`
F	Starts the request on the first day of the specified time unit (weekend, business week, week number, month number). For example, to start a request on the first Monday in November, use these FROM options: `from f mon nov`
L	Starts the request on the last day of the specified time unit (weekend, business week, week number, month number). For example, to start a request on the last Monday in November, use these FROM options: `from l mon nov`
dayname	Starts a request on a specific day of the week (MON, TUE, WED, THU, FRI, SAT, or SUN)
WE	Starts a request on a weekend day
W	Starts a request at the beginning (Monday) of the specified week. For example, to start a request the beginning of the third week in July, use these FROM options: `from 3 w jul`

Option	Meaning
B	Starts a request on a business day
weeknumber	Starts a request the nth week from the current week: 001 means next week, 002 means the week after that, and so on. For example, to start a request on Tuesday of next week if specified on Monday of the current week, use the following FROM options: `from 1 tue 001`
M	Starts a request on the specified day or week of the current month. For example, to start a request on the third week of this month, use these FROM options: `from 3 w m`
monthname	Starts a request on the specified day or week of the named month (such as JAN or FEB). For example, to start a request on the second weekend day in January, use these FROM options: `from 2 we jan`
Q	Starts a request on the specified day or week of the current quarter. For example, to start a request on the last business day of this quarter, use these FROM options: `from l b q`
*n*Q	Starts a request on the specified day or week of the nth quarter. For example, to start a request the first week of the fourth quarter, use these FROM options: `from f w 4q`

ON *day*

Specifies the day of the week on which the EXEC is scheduled to execute. Valid day values are MON, TUE, WED, THU, FRI, SAT, and SUN. The initial run date is calculated to fall on the next occurrence of *day*. EXECs scheduled for a specific day cannot be repeated using the AGAIN DAILY or AGAIN MONTHEND options; however, any other AGAIN scheduling intervals can be specified. You can use ON with the FOR option as a repeat factor.

RELEASE *userid*

Specifies this request as a dependent request. Requests scheduled with the RELEASE option are never explicitly scheduled for execution. Therefore, the server ignores all scheduling parameters except WITHIN. The user ID specified is a user ID that is allowed to release the request for initiation using the RELEASE command. Other user IDs can release the request, including the user ID of the person owning the request and a user ID having NOPASS authority. Also, a person knowing the password of the person owning the request can release the request.

WITHIN {*hh*:*mm*:*ss* | *}

Specifies the amount of time *CA VM:Schedule* is to keep trying to start the request if it cannot start at the scheduled time. If the EXEC does not start within this amount of time, it is skipped. Your site can define the default initiation time limit. If you specify *, the WITHIN time is unlimited.

Repeat Options

[AGAIN [BI] *interval*]

Specifies how often you want to run the request.

BI repeats the request every other interval. For example, AGAIN BI DAILY repeats a request every other day. BI is not valid with the HOURLY, YEARLY, *number* DAYS, or *number* BDAYS options.

The following table describes the intervals and purpose:

Interval	Purpose
HOURLY	Starts the request every hour (not valid with BI)
DAILY	Starts the request every day; for example, AGAIN DAILY starts a request every day; AGAIN [BI] DAILY starts a request every other day
BDAILY	Starts the request every business day
WEEKLY	Starts the request every week
WEEKEND	Starts the requests every weekend day
MONTHLY	Starts the request every month
QTRLY	Starts the request every quarter
YEARLY	Starts the request every year (not valid with BI)
MONTHEND	Starts the request on the last day of the month
QTREND	Starts the request on the last day of the quarter
number DAYS	Starts the request every number of days (not valid with BI)

Interval	Purpose
number BDAYS	Starts the request every number of business days (not valid with BI)

DURING *options*

Repeats the request on the requested day or days during the specified period. Options are:

nnn

Repeats a request on the requested days every *nnn*th week (Monday through Sunday). For example, again daily during three schedules a request to run daily for a week every three weeks.

M

Repeats a request on the specified days or weeks of the current month; for example, on mon during m repeats a request every Monday during this month.

monthname

Repeats a request on the specified days or weeks of the named month (such as JAN or FEB). For example, again daily during feb repeats a request every day during February.

Q

Repeats a request on the specified days, weeks, or months of the current quarter; for example, again weekly during q repeats a request every week during this quarter.

nQ

Repeats a request on the requested days, weeks, or months of the specified quarter; for example, again daily during 1q repeats a request every day during the first quarter.

[ON *day*[-*day*]]...

Specifies the day of the week on which the request is scheduled to execute. Valid day values are MON, TUE, WED, THU, FRI, SAT, and SUN. The initial run date is calculated to fall on the next specified day. When you use this option with the DURING option, the server repeats the request on the requested day or days during the specified period.

Requests scheduled for a specific day cannot be repeated using the AGAIN BDAILY, AGAIN DAILY, AGAIN MONTHEND, AGAIN WEEKEND, or AGAIN QTREND options; however, any other AGAIN scheduling intervals can be specified. You can use ON with FOR option as a repeat factor.

You can specify the ON option multiple times, for example, on mon on wed on fri.

You can use a dash to specify a range of days for example, mon-fri, sat-sun.

EVERY *hh*:*mm*:*ss*

Specifies the interval of time until the next initiation of a request on a given day. The maximum interval is 23:59:59 and can extend beyond midnight.

Resource Limit Options

CPU *seconds*

Specifies the threshold for the total (virtual plus overhead) CPU in seconds that the autologged virtual machine can consume. If this value is exceeded, the server calls the MONITOR user exit. The CPU option is effective only if your site has implemented a MONITOR user exit routine to monitor resources used by the server requests.

SIO *count*

Specifies the threshold for the disk and tape I/Os the autologged virtual machine can perform during request execution. If this value is exceeded, the server calls the MONITOR user exit. The SIO option is effective only if your site has implemented a MONITOR user exit routine.

UIO *count*

Specifies the threshold for the unit record (reader, printer, punch) I/Os the virtual machine can perform during request execution. If this value is exceeded, the server calls the MONITOR user exit. The UIO option is effective only if your site has implemented a MONITOR user exit routine.

STORAGE *size*

Specifies the storage size in kilobyte (K) or megabyte (M) units of the virtual machine that executes the request. The specified size must be equal to or less than the virtual machine's maximum storage size as currently set in the CP directory.

Initiation Limits Options

FOR *number*

Specifies the number of times that you want to run the request. Requests scheduled with the FOR option must also include an ON, AGAIN, or EVERY option.

TOTIME *hh*:*mm*:*ss*

Specifies the latest time of day at which a request scheduled to repeat with the EVERY option is to initiate.

UNTIL *mm/dd/yy*

Specifies a date that limits the automatic rescheduling of the request. The request will run up to and including the UNTIL date.

All dates that you specify with the UNTIL option must contain a month and a day. If you omit the year, the default is the current year. The latest date that you can specify is December 31, 2041. *CA VM:Schedule* interprets the years 00-41 as 2000-2041.

UNTIL *options*

Runs a request up to, and including a given day, or week of a particular week, month, or quarter. The following table presents the UNTIL options:

Option	Purpose
dayname	Runs the request until the end of the named day (MON, TUE, WED, THU, FRI, SAT, and SUN). For example, to run a request through the end of the first Monday in November use these UNTIL options: `until f mon nov`
WE	Runs the request until the end of the specified weekend day
W	Runs the request until the beginning of the specified week (through Monday). For example, to run a request through the end of Monday of the last week in July, use these UNTIL options: `until l w jul`
B	Runs the request until the end of the specified business day
weeknumber	Runs the request until the nnnth week (3-digit weeks) from this week. For example, if on Monday you want to schedule a request to run daily up to and including Tuesday of the following week, use these UNTIL options: `until 1 tue 001`

Option	Purpose
M	Runs the request until the end of the specified day or week of the current month. For example, to run a request through the end of the third week of this month, use these UNTIL options: `until 3 w m`
monthname	Runs the request until the end of the specified day or week of the named month (such as JAN or FEB). For example, to run a request through the second weekend day in January, use these UNTIL options: `until 2 we jan`
Q	Runs the request until the end of the specified day or week of the current quarter. For example, to run a request through the end of the last business day of this quarter, use these UNTIL options: `until l b q`
nQ	Runs the request until the end of the specified day or week of the nth quarter. For example, to run a request through the end of Monday of the last week of the fourth quarter, use these UNTIL options: `until l w 4q`

Shift and Range Options

INSIDE {*shift* | *range*}

Specifies a name associated with a time or date period that the *CA VM:Schedule* system administrator defined. (Use the CONFIG command to determine the shifts and ranges that your site defined.) The request does not run unless the time specified with the AT or FROMTIME option falls inside the specified shift or range. The INSIDE option cannot be specified with the OUTSIDE option.

If you specify an INSIDE shift or range without using an AT time or FROM date, *CA VM:Schedule* uses as the default the first available time and day within the specified shift or range.

OUTSIDE {*shift* | *range*}

Specifies a name associated with a date or time period that the *CA VM:Schedule* system administrator defined. (Use the CONFIG command to determine the shifts and ranges defined at your site.) The request does not run unless the time specified with the AT or FROMTIME option falls outside the specified shift or range. The OUTSIDE option cannot be specified with the INSIDE option.

If you specify an OUTSIDE shift or range without using an AT time or FROM date, *CA VM:Schedule* uses as the default the first available time and day not included in the specified shift or range.

Log File Options

EXCEPT [SPOOL *userid*]

Specifies that only exceptional events (such as skipping a request, or request execution errors) should be recorded in the message file sent to the requesting user ID after *CA VM:Schedule* initiates a request. This is the default.

SPOOL *user ID* specifies the user ID of the virtual machine that is to receive messages. The default user ID is the user ID of the virtual machine scheduling the request. The SPOOL option is ignored if you also specify the NOLOG option.

LOG [SPOOL *userid*]

Specifies that a message file is to be sent to the virtual machine specified by the SPOOL option whenever the request initiates. (If the SPOOL option is not specified, the message file is sent to the virtual machine that scheduled the request.) You need special authorization to use the LOG option.

SPOOL *userid* specifies the user ID of the virtual machine that is to receive messages. The default user ID is the user ID of the virtual machine scheduling the request. The SPOOL option is ignored if you also specify the NOLOG option.

NOLOG

Specifies that a message file is not produced. You need special authorization to use the NOLOG option.

Monitoring Options

MONITOR {YES | NO}

Specifies whether request monitoring is to be performed. *CA VM:Schedule* uses configuration file settings (or system defaults) if you specify only some or no other monitoring options. If not specified in the VMSCHED CONFIG file, the default is NO. If MONITOR YES is specified in the VMSCHED CONFIG file, you cannot use the MONITOR NO option.

LOGOff {YES | NO}

Specifies whether to log off the requesting user ID when the request ends. Specifying LOGOFF YES automatically invokes request monitoring. If you specify NO, the user ID remains logged on. If not specified in the VMSCHED CONFIG file, the default is NO.

CONSOLE {YES | NO}

Spools the requesting user ID's console. If not specified in the VMSCHED CONFIG file, the default is NO. Specifying this option automatically invokes request monitoring.

CONSTO *userid*

Specifies the user ID to spool the console to. The default is the requesting user ID. Specifying the CONSTO option automatically invokes request monitoring.

DUMP {CP | VM | N}

Performs a CP or VM dump if the request abends. N specifies that no dump is to be generated. Specifying DUMP CP or DUMP VM automatically invokes request monitoring. If not specified in the VMSCHED CONFIG file, the default is N.

DUMPTO *userid*

Specifies the user ID to receive the dump. The default is the requesting user ID. The DUMPTO option cannot be used with DUMP N. Specifying the DUMPTO option automatically invokes request monitoring. If the type of dump is not specified by the user or in the VMSCHED CONFIG file, DUMP VM is assumed.

Miscellaneous Options

PASSWORD *password*

Specifies the CP LOGON password of the virtual machine being scheduled. If you do not specify a password or specify a **?**, *CA VM:Schedule* prompts for the password. If the SCHEDULE command is invoked by means of the CP SMSG command, you must specify the password explicitly on the command line.

Your *CA VM:Schedule* system administrator can specify that users not be allowed to enter the password in the options string, but instead must be prompted for the password, in which case CP SMSG cannot be used. *CA VM:Schedule* verifies your CP LOGON password when you issue the SCHEDULE command. You can change your password without affecting future initiations of the request.

SYSNAME {*systemname* | *}

Specifies the system name of the Single System Image member where this request is to run. The system name specified must be a member of the SSI cluster where *CA VM:Schedule* runs. Specify * if you want to clear this field in a request that has it already set. If not specified the request is run on the SSI member where *CA VM:Schedule* is running. This option is invalid if *CA VM:Schedule* is not configured for Single System Image mode.

USER *userid*

Specifies the user ID on which the request is to run. If not specified, the user ID defaults to the user ID that issued the SCHEDULE command. When you schedule a request to run on another user ID, make sure the other user ID either owns the scheduled program or EXEC, or has access to it.

Description

You can also use CHANGE from *CA VM:Batch*, through the *CA VM:Schedule* interface to *CA VM:Batch*, to change scheduled *CA VM:Batch* jobs. For complete information about using the *CA VM:Schedule* interface to *CA VM:Batch*, refer to the *CA VM:Batch User and Group Manager Guide*.

Note: You can use the CHANGE command to nullify a CANCEL command that is issued against a request. Run the CHANGE command on the request. Make an arbitrary change. The CHANGE command removes the CANCEL flag.

To keep any resulting console spool files free from extraneous data, make sure that your PROFILE EXECs and scheduled EXECs contain a SET BLIP OFF command.

If you have any options that are set to a value, you cannot nullify that value with the line mode CHANGE command. For example, if you set SHIFT to FIRST, you cannot set SHIFT to blanks from line-mode. You must go into full-screen mode.

Example

You have scheduled a request named STATUS to run a long-running program named REPORTA every weekend while you are not in the office. Your original SCHEDULE command was the following line:

```
vmsched schedule reporta (at 21:00:00 on fri again weekly
```

You decide to run the request daily, and you want the user ID logged off, and to have *CA VM:Schedule* monitor the outcome. To make those changes to the request named STATUS, enter this command:

```
vmsched change status (again daily logoff yes monitor yes
```

Duplicate Options

If you specify duplicate options on the command line, *CA VM:Schedule* uses the last option on the line; all previous option entries are ignored.

Note: This behavior does not apply to the ON option.

Resource Limits

If your site implemented a MONITOR user exit routine to monitor the resources that your server requests use, the resources that you specify as options in the CHANGE or SCHEDULE command are monitored. For details about resource monitoring, see your *CA VM:Schedule* system administrator.

The CONFIG command displays the default values for CPU, SIO, and UIO, but only if the server monitors resource consumption for that parameter.

Your site can specify resource limits for classed requests. You can display these limits with the CONFIG command, even if a MONITOR user exit has not been implemented at your site. However, in this case resources are not monitored.

The resource limits for requests can be set in three locations:

- The USEREXIT MONITOR record. These limits (if set) are the default resource limits.
- The CLASS record (classed requests only). These limits are the maximum resource limits for the request processing class. These limits override any maximum limits that are set on the USEREXIT MONITOR record.
- Options on the SCHEDULE command. Limits that are set here override any maximum limits that are set on the USEREXIT MONITOR record. For classed requests, limits that are set here cannot exceed the limits (if any) set in the CLASS record. However, they can be more restrictive than those limits.

Request Logging

The server sends a spool file to the user ID scheduling a request when a request is initiated, unless you specify the NOLOG option. When you schedule a request with the FOR option, only one spool file is sent for the last successful initiation of the request.

If you do not specify either the LOG or the NOLOG option, the server sends a message file only when exceptional conditions occur. Exceptional conditions include the skipping of a request initiation due to unsuccessful autolog attempts, the final run of a request, and the occurrence of any type of request execution error. To determine if you are authorized to use the LOG or NOLOG option, check with your *CA VM:Schedule* system administrator.

When Requests Fail

If the virtual machine is logged on or another autolog error occurs, requests fail to initiate. If the failure occurs because the virtual machine is logged on, the initiation must begin within the WITHIN period, or the request is skipped. If the failure is because of any other autolog error, the request is canceled.

Restrictions

You cannot use the CHANGE command to change an EXEC belonging to another user ID. If you must modify someone else's EXEC, see your system administrator and request that ownership of the EXEC be transferred to your user ID. Then modify the EXEC from your user ID. When you finish, have the system administrator transfer ownership of the EXEC back to the original user ID.

CONFIG Command

The CONFIG command displays the *CA VM:Schedule* configuration settings that your site implemented. Contact your *CA VM:Schedule* system administrator if you have questions about a particular configuration setting in effect at your site.

```
CONFIG [EXEC]
```

Definitions

EXEC

Displays the names of any EXECs on the *CA VM:Schedule* minidisks you are authorized to schedule with the EXEC command.

Description

The CONFIG command displays the *CA VM:Schedule* configuration settings that are implemented at your site.

Example

To display the local scheduling parameters that are implemented at your site, enter the following command:

```
vmsched config
```

Example

To list EXECs on the *CA VM:Schedule* minidisks that you are authorized to use, enter the following command:

```
vmsched config exec
```

DELAY Command

The DELAY command modifies the scheduled initiation time of a request to a later time, preventing *CA VM:Schedule* from interrupting an active terminal session.

```
DELay requestname {UNTIL hh:mm:ss | FOR hh:mm:ss} [("Options"]


Options:

[User userid]
[Password password]
```

Definitions

requestname

Specifies the name of the request being delayed.

UNTIL *hh*:*mm*:*ss*

Specifies the time of day that you want the rescheduled request to initiate. The time that the UNTIL parameter specifies is used for all subsequent initiations of the request. By using the DELAY command with the UNTIL parameter, you can schedule a request or series of requests to initiate earlier than the original scheduling command specified. To complete this task, specify an UNTIL time between the present time and the next scheduled initiation of the request. Any subsequent initiations of the request are based on this new, earlier time.

FOR *hh:mm:ss*

Specifies the amount of time that you want to delay a request. The delay interval that the FOR option specifies affects only the next initiation of the request. If the request to be delayed has already attempted to initiate, the initiation time of the request delayed using FOR is calculated by adding the delay time interval to the time the DELAY command is issued. Subsequent initiations of the request are not affected.

USER *userid*

Specifies the user ID whose request you are going to delay. If you do not specify a user ID, the default is your user ID.

Password *password*

Specifies the CP LOGON password of the user ID whose request you are going to delay.

Description

The DELAY command changes the scheduled initiation of a request time to a later time, preventing *CA VM:Schedule* from interrupting an active terminal session.

Example

You schedule a request, TASK1, to start now, but you are using your virtual machine. The server sends you the following warning message:

```
ATTEMPTING TO INITIATE REQUEST 'TASK1'.
```

To delay this request for 5 minutes, enter the following command:

```
vmsched delay task1 for 00:05:00
```

Example

You schedule request TASK3 to execute at 9:00 a.m. every day from user ID LUCY. You decide to execute the request every day at 8:00 a.m. instead. If it is before 8:00 a.m., enter the following command:

```
vmsched delay task3 until 08:00:00 (user lucy
```

TASK3 runs at 8:00 a.m. (one hour earlier than it was originally scheduled to run) every day from now on.

EXEC Command

The EXEC command schedules an EXEC to execute on the *CA VM:Schedule* service virtual machine. The *CA VM:Schedule* system administrator must authorize you explicitly to execute an EXEC on the virtual machine. To list EXECs that you are authorized to use, enter the command **vmsched config exec.**

```
Exec requestname filename [execparms] [( "Options"]


Options:

["Initial date/time options"]
["Repeat options"]
["Initiation limits options"]
["Shift and range options"]
["Log file options"]
```

```
Initial date/time options:

[AT hh:mm:ss]
[FROMTime hh:mm:ss]
[FRom {"From options"}]
[ON day[-day]]...
[Within {hh:mm:ss | *}]


From options:

  mm/dd/yy

| dayname

| [“ParmA” [“ParmB” [“ParmC”]]]


Repeat options:

[AGain [BI] Interval]

[On day [-day]]

[Every hh:mm:ss]

[DURing {nnn | M | monthname | Q | nQ}]


Initiation limits options:

[For number]
[Totime hh:mm:ss]
[UNtil {"Until options"}]

Until options:

  mm/dd/yy
| dayname
| [“ParmA” [“ParmB” [“ParmC’]]]


ParmA:

{nnn | F | L}


ParmB:

{dayname | WE | W | B}
```

```
ParmC:

{weeknumber | M | monthname | Q | nQ}


Shift and range options:

[Inside {shift | range}]
[OUtside {shift | range}]


Log file options:

[EXCept [SPool userid]]
[Log [SPool userid]]
[Nolog]
```

Definitions

requestname

Specifies the name that you give the request, from one to eight characters long. You cannot already have an EXEC or a request that you scheduled with this name.

filename

Specifies the filename of an EXEC residing on a *CA VM:Schedule* minidisk to be executed on the *CA VM:Schedule* service virtual machine. If no parameters are specified on the command line, the server prompts you for them. In a prompted entry, you can use parentheses. The maximum parameter string length is 108 characters, including the filename, punctuation, and blanks.

execparms

Specifies any parameters that the EXEC requires to execute properly. The parameters cannot contain parentheses.

Initial Date/Time Options

AT *hh*:*mm*:*ss*

Specifies the time of day at which the EXEC is to initiate. If you do not specify AT nor FROMTIME is specified, the time of day defaults to a site-specified delay interval from the moment the command is issued. See *Start Times* (see page 97) for more information about when requests are scheduled to start.

FROMTIME *hh*:*mm*:*ss*

Specifies the time of day at which the EXEC is to initiate. Use FROMTIME with TO to specify a time period that spans midnight. See *Start Times* (see page 97) for more information about when requests are scheduled to start.

FROM *mm*/*dd*/*yy*

Specifies the date of the first (or only) EXEC initiation. If you do not specify the FROM option, the first initiation defaults to the current date, unless you specify that initiation is to occur on certain days of the week, or if the specified time has passed for the current day.

Note: All dates that you specify with the FROM option must contain a month and a day. If you omit the year, the default is the current year. The latest date that you can specify is December 31, 2041. Year ranges from 00 through 41 are taken to be in the 21st century.

FROM *dayname*

Specifies the day of the first (or only) EXEC initiation. If *dayname* is today, the server schedules the request to run a week from today. If you do not specify the FROM option, the first initiation defaults to the current date, unless you specify that initiation is to occur on certain days of the week, or if the specified time has passed for the current day.

FROM "From options"

Schedules the first or only initiation of a request on a given day, or week of a particular week, month, or quarter. From options are the same as the FROM option on the CHANGE and SCHEDULE commands.

WITHIN {*hh*:*mm*:*ss* | *}

Specifies the amount of time that the server keeps trying to start the request, if it cannot start at the scheduled time. If the EXEC does not start within this amount of time, it is skipped. Your site can define the default initiation time limit. If you specify *****, the WITHIN time is unlimited; your request has to run.

[ON *day*[-*day*]]...

Specifies the day of the week on which the EXEC is scheduled to execute. Valid day values are MON, TUE, WED, THU, FRI, SAT, and SUN. The initial run date is calculated to fall on the next occurrence of *day*. When you use this option with the DURING option, CA VM:Schedule repeats the EXEC on the requested day or days during the specified period.

EXECs scheduled for a specific day cannot be repeated using the AGAIN DAILY or AGAIN MONTHEND options; however, you can specify any other AGAIN scheduling intervals. You can use ON with FOR as a repeat factor.

You can specify the ON option multiple times, for example, on mon on wed on fri.

You can use a dash to specify a range of days, for example, mon-fri, sat-sun.

Repeat Options

AGAIN [BI] *interval*

Specifies how often you want to execute the EXEC. Intervals are the same as the AGAIN [BI] interval on the CHANGE command.

BI repeats the request every other interval. For example, BI DAILY repeats a request every other day. You cannot use BI with the HOURLY, YEARLY, *number* DAYS, or *number* BDAYS options.

DURING *option*

Repeats the request on the requested day or days during the specified period. Options are the same as the DURING option on the CHANGE command.

[ON *day*[-*day*]]...

Specifies the day of the week on which the new EXEC is scheduled to execute. When you use this option with the DURING option, the server repeats the request on the requested day or days during the specified period. Valid day values are MON, TUE, WED, THU, FRI, SAT, and SUN. You can use a dash to specify a range of days (for example, MON-FRI, SAT-SUN). You can also specify the ON option multiple times (for example, ON MON ON WED ON FRI). The initial run date is calculated to fall on the next specified day. EXECs that you schedule for a specific day cannot be repeated using the AGAIN DAILY or AGAIN MONTHEND options. However, you can specify any other AGAIN scheduling intervals. You can use ON with FOR as a repeat factor.

EVERY *hh:mm:ss*

Specifies the interval of time until the next initiation of the EXEC on a given day. The maximum interval is 23:59:59 and can extend beyond midnight.

Initiation Limits Options

FOR *number*

Specifies the number of times to run the request. EXECs scheduled with the FOR option must also include an ON, AGAIN, or EVERY option.

TOTIME *hh*:*mm*:*ss*

Specifies the latest time of day at which an EXEC scheduled to repeat with the EVERY option is to initiate.

UNTIL *mm/dd/yy*

Specifies a date that limits the automatic rescheduling of the EXEC. The request runs up to, and including the UNTIL date.

All dates you specify with the UNTIL option must contain a month and a day. If you omit the year, the default is the current year. The latest date that you can specify is December 31, 2041. The server interprets the years 00-41 as 2000-2041.

UNTIL *dayname*

Specifies a day that limits the automatic rescheduling of the EXEC. The request runs up to, and including the UNTIL day.

Shift and Range Options

INSIDE {*shift* | *range*}

Specifies a name associated with a time or date period defined by the *CA VM:Schedule* system administrator. Use the CONFIG command to determine the shifts and ranges defined at your site.

The EXEC will not run unless the time specified with the AT or FROMTIME option falls inside the specified shift or range. The INSIDE option cannot be specified with the OUTSIDE option.

If you specify an INSIDE shift or range without using an AT time or FROM date, the server uses as the default the first available time and day within the specified shift or range.

OUTSIDE {*shift* | *range*}

Specifies a name associated with a date or time period, defined by the *CA VM:Schedule* system administrator. Use the CONFIG command to determine the shifts and ranges defined at your site.

The EXEC will not run unless the time specified with the AT or FROMTIME option falls outside the specified shift or range. The OUTSIDE option cannot be specified with the INSIDE option.

If you specify an OUTSIDE shift or range without using an AT time or FROM date, the server uses as the default the first available time and day not included in the specified shift or range.

Log File Options

EXCEPT [SPOOL *userid*]

Specifies that only exceptional events (such as skipping a request, or request execution errors) should be recorded in the message file sent to the requesting user ID after the server initializes a request. This option is the default.

SPOOL *userid* specifies the user ID of the virtual machine that is to receive punched messages. The default user ID is the user ID of the virtual machine scheduling the EXEC. The SPOOL option is ignored if you also specify the NOLOG option.

LOG [SPOOL *userid*]

Specifies that a message file is to be sent to the virtual machine specified by the SPOOL option whenever the EXEC initiates. (If the SPOOL option is not specified, the message file is sent to the virtual machine that scheduled the EXEC.) You need special authorization to use the LOG option.

SPOOL *userid* specifies the user ID of the virtual machine that is to receive punched messages. The default user ID is the user ID of the virtual machine scheduling the EXEC. The SPOOL option is ignored if you also specify the NOLOG option.

NOLOG

Specifies that a message file is not produced.

Description

The EXEC command schedules an EXEC to execute on the *CA VM:Schedule* service virtual machine.

Example

You want to send user ID JOE a message at 11:45 a.m., Monday through Friday, every week to remind your friend that it is lunch time. To schedule an EXEC request named EAT that executes the VMDMSG EXEC on the *CA VM:Schedule* service virtual machine to send the message, enter the following command:

```
vmsched exec eat vmdmsg joe lunch time (on mon-fri at 11:45 again weekly
```

Example

To schedule a request named RUNREPT to run the REPORT EXEC every Monday of the current quarter, enter the following command:

```
vmsched exec runrept report (on mon during q
```

Example

To schedule a request named CHECKUP that runs the CHECKUP EXEC through January, beginning January 10, enter the following command:

```
vmsched exec checkup checkup (again daily during jan from 01/10/yy
```

Example

You have an EXEC named MBACKUP that submits your monthly backups. To schedule a request named BACKUP that runs the MBACKUP EXEC on the first Saturday of every month at 11 p.m., enter the following command:

```
vmsched exec backup mbackup (at 23:00 from f sat m again monthly
```

Start Times

If you do not specify a start time by using the AT or FROMTIME options, the start time defaults to a site-specified delay interval from the moment the command is issued, not completed. If your site's delay interval is short, *CA VM:Schedule* might skip an EXEC's first initiation if you take a long time to reply to prompts for command parameters or passwords.

The delay interval applies only to requests scheduled to run today. For example, suppose your system delay time is five minutes. If at 11 a.m. you schedule a request to run tomorrow without specifying a start time, *CA VM:Schedule* will schedule the request to run at 11 a.m. tomorrow, not 11:05 tomorrow. Use the CONFIG command to find the default interval used at your site.

Scheduling CA VM:Schedule Commands

If the EXEC being executed on the *CA VM:Schedule* service virtual machine contains *CA VM:Schedule* commands, use the following format:

```
cp smsg * command [parameters] [(options]
```

- EXECs that require modification of user storage cannot be scheduled to execute on the *CA VM:Schedule* service virtual machine.
- If you specify duplicate options on the command line, the server uses the last duplicate option on the line, and it ignores all previous duplicate option entries. This behavior does not apply to the ON option.

- When EXECs are executed on an operational *CA VM:Schedule* system, the server stacks the user ID and then the EXEC parameters. If the EXEC on the *CA VM:Schedule* service virtual machine must use the stack, a "read" of the first stacked line or a DESBUF must be performed at the beginning of the EXEC.
- *CA VM:Schedule* commands issued through the CP SMSG command must have all parameters and passwords specified in the command line because the server is unable to prompt for them.

HOLD Command

The HOLD command suspends or resumes request initiations. A request remains on hold until you remove it.

```
Hold {ON | OFF} requestname [([User userid] [Password password]]
```

Definitions

ON

Specifies that the named request is to be placed on hold. This cannot be specified with pattern matching.

OFF

Specifies that the named request is to have the hold removed. This cannot be specified with pattern matching.

requestname

Specifies the name of the request to be placed on hold or released from hold.

USER *userid*

Specifies the user ID whose request you are going to place on hold. This cannot be specified with pattern matching.

PASSWORD *password*

Specifies the CP LOGON password for the user ID whose request you are going to place on hold.

Description

The HOLD command suspends or resumes request initiations. An on-hold request remains on hold until you remove the hold. If a request is on hold at its scheduled initiation time, the server does not initiate it, regardless of the specified WITHIN time. However, if the status of the request is changed before your system's WITHIN time (default late initiation time) passes, the server tries to initiate the request during the WITHIN period. The HOLD command can be applied to requests owned by other user IDs if the password is provided or if you have NOPASS authorization.

Consider the following information:

- The request remains on hold until the hold is removed.
- A request that is on hold cannot be skipped or delayed. All other functions can be performed.
- If an on-hold request is copied, the new request is not on hold and is treated as a new request.
- A request that attempts to initiate cannot be placed on hold.

Requests Automatically Held

The server places requests on hold automatically under the following circumstances:

- An autolog error terminates the initiation.
- During initiation of an EXEC on the server, the EXEC to be executed is not found on any of the disks accessed by the service virtual machine.
- Using the EXEC command, you scheduled an EXEC that you are no longer authorized to use.
- The logon storage size specified in the storage scheduling option now exceeds the maximum allowed for that user ID.

QUERY Command

The QUERY command displays information about one or more scheduled requests. The display is either in long or short form. See the descriptions of the SHORT and LONG options that follow for details. You cannot issue this command cannot with the CP SMSG VMSCHED command.

```
Query  [requestname | *] [( "Options"]


Options:

["Output destination option"]
["Output length option"]
["Miscellaneous options"]


Output destination options:

[File filename] [PRint] [STack {FIFO | LIFO}] [Term]


Output length options:

[Long | Short]


Miscellaneous options:

[LAst | NExt]
[Password password]
[SYSNAME systemname]
[User userid]
```

Definitions

requestname

Specifies the name of a *CA VM:Schedule* request you want to display. The default is all requests (specified by an asterisk).

Output Destination Options

The FILE, PRINT, STACK, and TERM options can be specified together on a single QUERY command line. Only the TERM option can be specified when entering the QUERY command from the *CA VM:Schedule* console. If none of the FILE, PRINT, STACK, or TERM options is specified, the output is displayed on your terminal.

FILE *filename*

Writes the output of the QUERY command to the file *filename* VMSCHED A1. If you do not specify the FILE option, the output is displayed on your terminal.

PRINT

Sends the output from the QUERY command to your virtual printer. If the PRINT option is not specified, the output displays on your terminal.

STACK {FIFO | LIFO}

Specifies that output is to be stacked FIFO (first in, first out) or LIFO (last in, first out) in your program stack. The option can be used when the QUERY command is being executed from within an EXEC. These options cannot be specified from full-screen mode.

TERM

Displays the output on your terminal. TERM is the default if no other output option is specified.

Output Length Options

LONG

Displays a more detailed form of the QUERY display format.

SHORT

Displays an abbreviated form of the QUERY display format. This option displays a single line containing the request name, execution status, last execution date, next execution date, and the owning user ID for each request. If *CA VM:Schedule* is running in Single System Image mode, the system name is also displayed in the output. SHORT is the default QUERY display format.

Miscellaneous Options

LAST | NEXT

Displays the last initiated request for the specified user ID (for LAST) or the next scheduled request for the specified user ID (for NEXT). If you do not specify a user ID, the user ID that issued the QUERY command is the default.

PASSWORD *password*

Specifies the CP LOGON password for the user ID whose requests are being queried. You must specify the password when you query the requests of another user ID. Your system administrator can specify that users are not allowed to enter the password in the option string. Instead, you are required to use the prompting procedure for password entry. You do not have to supply your password when querying requests you submitted on your own user ID.

SYSNAME {*systemname* | *}

Specifies the system name of a Single System Image member. If specified, only requests that are defined to run on this member, and that meet the rest of the selection criteria, display. Specify * if you want to only QUERY the requests that are not assigned a SYSNAME. If not specified, then *systemname* is not used as part of the output selection criteria. This option is invalid if *CA VM:Schedule* is not configured for Single System Image mode.

USER *userid*

Specifies the user ID whose requests you want to query. The default is the user ID that issued the QUERY command. If you specify another user ID, the CP logon password must be specified unless you have NOPASS authorization. If you have NOPASS authorization, you can specify an asterisk to query requests for all user IDs.

Description

The QUERY command displays information about one or more scheduled requests. The display is either in long or short form.

Example

To query the status of all requests with names that begin with PROD and that belong to user LYDIA, enter the following command:

```
vmsched query prod* (user Lydia
```

CA VM:Schedule displays the following response:

```
ENTER PASSWORD:
```

Enter your password.

If *CA VM:Schedule* recognizes it as the correct password, it displays output similar to the following output:

```
REQUEST EXECUTION STATUS LAST DATE/TIME        NEXT DATE/TIME        USERID
------- --------------- --------------------- --------------------- ------
PROD1   NORMAL EXECUTION WED 01/15/16 23:00:00 THU 01/16/15 23:00:00 LYDIA
```

Example

The long format of the QUERY display contains fields that represent the *CA VM:Schedule* scheduling capabilities and features. To query the status of request REQ1, enter the following command:

```
vmsched query req1 (long
```

CA VM:Schedule displays the following response:

```
USERID: POTTER                  REQUEST: REQ1           AT: 23:00:00

WITHIN: 00:02:00                EVERY: N/A              FOR: N/A

TO: N/A                         FROM: 01/15/14          UNTIL: 01/30/14

ON DAYS: N/A

SHIFT/RANGE: N/A

REPEAT INTERVAL: EVERY BUSINESS DAY


RELEASE BY: N/A                 SPOOL TO: POTTER        CLASS: N/A

STORAGE: N/A

REQUEST MONITOR: YES            SPOOL CONSOLE: YES      DUMP TYPE: VM

SEND CONSOLE TO: POTTER         SEND DUMP TO: POTTER    LOGOFF: YES

LAST RUN: WED 01/15/02 23:00:00 END TIME: WED 01/15/02 23:05:25  RC: 0

NEXT RUN: MON 01/16/02 23:00:00 STATUS: NORMAL INITIATION

COMMAND: EXEC DLYRPT

RESOURCE LIMITS:

CPU: 10                         SIO: 1000               UIO: 1000

RESOURCE USAGE:

CPU: 0                          SIO: 164                UIO: 0

ELAPSED RUN TIME: 00:05:25
```

Output Field and Content

USERID

Specifies the user ID that owns the request.

REQUEST

Specifies the name assigned to the request by the owner.

SYSNAME

Specifies a Single System Image member name the request runs on. If not assigned to a specific system in the SSI cluster, this field is * and the request runs on the single system image member where the *CA VM:Schedule* service machine runs.

AT

Specifies the time at which the request is scheduled to run.

WITHIN

Specifies the maximum amount of time within which the server attempts to run the request.

EVERY

Specifies the frequency of initiation of this request on a given day.

FOR

Specifies the number of times that the request runs. The FOR field corresponds with the MAXIMUM INITIATIONS field on SCHEDULE screen 3 and EXEC screen 3.

TO

Specifies the latest time of day at which a request scheduled with a repeat time interval runs. The TO field corresponds to the DON'T RESCHEDULE AFTER field on SCHEDULE screen 2 and EXEC screen 2.

FROM

Specifies the first date to run the request. The server calculates this date when it is implied by a descriptive day specification. The FROM field corresponds to the FROM date in the FIRST RUN OPTIONS on SCHEDULE screen 1 and EXEC screen 1.

UNTIL

Specifies the last date to run the request. The server calculates this date when it is implied by a descriptive day specification. The UNTIL field corresponds to the FINAL RUN OPTIONS on SCHEDULE screen 3 and EXEC screen 3.

ON DAYS

Specifies the days of the week on which the request is to execute.

SHIFT/RANGE

Specifies the name of a site-defined range of dates or times inside (or outside) of which the request has been scheduled to initiate.

REPEAT INTERVAL

Specifies the days or dates on which the request is to run in a given month (or every nth week).

RELEASE BY

Specifies the user ID that the scheduling user ID has authorized to run the request.

SPOOL TO

Specifies the user ID of a virtual machine that is to receive spooled messages about the request.

CLASS

Specifies the execution class under which the request was scheduled.

STORAGE

Specifies the virtual machine storage size in which the request is to be executed.

REQUEST MONITOR

Specifies whether the server uses request execution monitoring.

SPOOL CONSOLE

Specifies whether to spool the console of the user.

TYPE DUMP

Specifies the type of dump (CP DUMP or VMDUMP) to be taken if an abend occurs.

SEND CONSOLE TO

Specifies the user ID to which the console listing sends.

SEND DUMP TO

Specifies the user ID that receives any abend dumps that the request may produce.

LOGOFF

Specifies whether to log off the user ID automatically after the initiated request completes.

LAST-RUN

Displays the day, date, and time at which the request was last initiated. If an asterisk follows the time, it means the RELEASE command initiated the last run of this scheduled request.

NEXT-RUN

Specifies the day, date, and time of the next scheduled request initiation.

STATUS

Displays a summary of the results of the last initiation attempt. See Listing Request Information (see page 48) for a description of the possible statuses.

COMMAND

Specifies the command to execute, if the request was scheduled using the SCHEDULE command.

If the request was scheduled using the EXEC command, the COMMAND field is labeled SYS-EXEC instead of COMMAND, and the name of the EXEC and its parameters (if any) display.

RESOURCE LIMITS:

CPU

Specifies the maximum number of real CPU seconds (TTIME) that the virtual machine can use while executing the request.

SIO

Specifies the maximum number of disks and tape I/Os that the virtual machine can perform while executing the request.

UIO

Specifies the maximum number of unit records (reader, printer, punch) I/Os that the virtual machine can perform while executing the request.

END TIME

Displays the time at which the request completed.

RC

Displays the return code with which the request ended.

RESOURCE USAGE:

CPU

Displays the number of real CPU seconds (TTIME) that the virtual machine used while executing the request.

SIO

Displays the number of disks and tape I/Os that the virtual machine performed while executing the request.

UIO

Displays the number of unit records (reader, printer, punch) I/Os that the virtual machine performed while executing the request.

ELAPSED TIME

Displays the amount of time that the request took to run.

RELEASE Command

The RELEASE command causes immediate initiation of a scheduled or dependent request.

```
RELease userid requestname [(PASSWORD password]
```

Definitions

userid

Specifies the user ID of the virtual machine owning the request to be initiated.

requestname

Specifies the name of the request to be initiated.

PASSWORD *password*

Specifies the CP LOGON password of the virtual machine that the *user ID* parameter specified. Your *CA VM:Schedule* system administrator can specify that users are not allowed to enter the password in the option string. If your system operator enables this restriction, you use the prompting procedure for password entry.

You do not have to supply your password under the following circumstances:

- You release a request that you submitted from your own user ID
- You release a dependent request and your user ID was specified as the *release by* user ID when the request was created
- You release a request that submitted under a different user ID, but your system administrator has given you NOPASS authority.

Description

The RELEASE command lets you initiate a scheduled or dependent request immediately. A released request attempts to initiate until it exceeds the WITHIN time, after which it skips.

You can use the RELEASE command to chain together requests with order-dependent execution. See the *Running Requests That Depend on Other Requests* (see page 38) section for more information.

Example

User LYDIA has scheduled the request PROD1 with the RELEASE option:

```
vmsched sched prod1 acctrept (release lucy
```

When it is time for PROD1 to be released, LUCY can issue the following command:

```
vmsched release lydia prod1
```

SCHEDULE Command

The SCHEDULE command schedules a program, EXEC, CP, or CMS command that initiates on an autologged virtual machine.

```
Schedule requestname [command] [( "Options"]

Options:
["Initial date/time options"] [Within {hh:mm:ss | *}]
["Repeat options"]
["Resource limit options"]
["Initiation limits options"]
["Shift and range options"]
["Log file options"]
["Monitoring options"]
["Miscellaneous options"]
```

```
Initial date/time options:

  Class class
| Release userid [Within {hh:mm:ss | *}]
| “Event Time Options” [Within {hh:mm:ss | *}]


Event Time Options:

[AT hh:mm:ss]
[FROMTime hh:mm:ss]
[On day]
[FRom “From options”]


From options:

  mm/dd/yy
| dayname
| [“ParmA” [“ParmB” [“ParmC”]]]


ParmA:

{nnn | F | L}


ParmB:

{dayname | WE | W | B}


ParmC:

{weeknumber | M | monthname  | Q | nQ}

Repeat options:

[AGain [BI] interval]
[ON day[-day]]...
[Every hh:mm:ss]
[DURing {nnn | M | monthname | Q | nQ}]

Resource limit options:

[CPu seconds]
[Sio count]
[Uio count]
[STorage size]

Initiation limits options:

[For number]
```

```
[Totime hh:mm:ss]
[UNtil "Until options"]


Until options:

  mm/dd/yy

| dayname
| [“ParmD” [“ParmE” [“ParmF”]]]


ParmD:

{nnn | F | L}


ParmE:

{dayname | WE | W | B}


ParmF:

{weeknumber | M | monthname | Q | nQ}


Shift and range options:

[Inside {shift | range}]
[OUtside {shift | range}]


Log file options:

[EXCept [SPool userid]]
[Log [SPool userid]]
[Nolog]


Monitoring options:

[Monitor {Yes | No}]
[LOGOff {Yes | No}]
[CONSole {Yes | No}]
[CONSTo userid]
[DUMP {CP | VM | N}]
[DUMPTo userid]


Miscellaneous options:
```

```
[Password password]

[SYSname {systemname | *}]
[USer userid]
```

Definitions

requestname

Specifies the name of the request that you want to schedule. You must not already have a scheduled request with this name. However, you can reuse a request name after the original request is no longer scheduled to run. For example, the request has been canceled, skipped, or has already run for the last time.

command

Specifies the EXEC, program, CP command, or CMS command that executes when the request runs. If specified on the *CA VM:Schedule* command line, the command parameters cannot contain parentheses. If you do not specify a command, the server prompts for it. (In a prompted entry, parentheses can be specified.) The maximum command parameter length is 108 characters, including filename, blanks, and punctuation.

The command line is passed to the console stack of the user ID when that user ID is autologged.

The PROFILE EXEC of any virtual machine that the server autologs must not contain any commands that alter or destroy the line that the server places in the console input buffer (sometimes referred to as the console stack). If this issue occurs, or the first read to the console is issued prematurely, the scheduled request might not execute as expected, even though the virtual machine is autologged successfully. For a more detailed discussion of the console stack, refer to the CMS user's guide appropriate for your site.

The PROFILE EXEC of any virtual machine that the server autologs should not prompt for parameters when the virtual machine is running disconnected.

Initial Date/Time Options

AT *hh*:*mm*:*ss*

Specifies the time of day to initiate the request. If you do not specify AT nor FROMTIME, the time of day defaults to a site-specified delay interval from the moment you issued the command. See *Start Times* (see page 97) for more information about when requests are scheduled to start.

CLASS *class*

Specifies the initiation class for the request. *class* is the one-character class identifier under which the command is to operate. Valid class values are A-Z and 0-9, and your system administrator defines these values. You cannot use this option with the following options:

- AGAIN HOURLY
- AT *hh:mm:ss*
- EVERY *hh:mm:ss*
- FROMTIME hh:mm:ss
- INSIDE range
- OUTSIDE range
- RELEASE userid
- TOTIME hh:mm:ss
- WITHIN hh:mm:ss

Note: Requests scheduled with the CLASS option are initiated by a *CA VM:Schedule* operator or through a schedule determined by your site. You can reschedule classed requests, but not more than once per day. To determine what classes (and resource limits per class) have been specified at your site, use the CONFIG command.

Note: For information about the request processing classes that correpond to the SCHEDULE CLASS option, see "CLASS: Defining Classes" in the *CA VM:Schedule Administration Guide*.

FROMTIME *hh*:*mm*:*ss*

Specifies the time of day at which the EXEC is to initiate. Use FROMTIME with TO to specify a time period that spans midnight. If FROMTIME is used with AT time, the FROMTIME option takes precedence. See *Start Times* (see page 97) for more information about when requests are scheduled to start.

FROM *mm/dd/yy*

Specifies the date of the first (or only) request initiation. If the FROM option is not specified, the first initiation defaults to the current date unless you specify that initiation is to occur on certain days of the week or the specified time has passed for the current day.

All dates that you specify with the FROM option must contain a month and a day. If you omit the year, the default is the current year. The latest date that you can specify is December 31, 2041. Year ranges from 00 through 41 are taken to be in the 21st century.

FROM *dayname*

Specifies the day of the first (or only) request initiation. If *dayname* is today, the server schedules the request to run a week from today. If the FROM option is not specified, the first initiation defaults to the current date unless you specify that initiation is to occur on certain days of the week or the specified time has passed for the current day.

FROM *options*

Schedules a request's first, or only initiation on a given day, or week of a particular week, month, or quarter. The following table describes the options and meanings:

Option	Meaning
nnn	Starts the request on the nth day or week. For example, to start a request on the second Monday in January, use these FROM options: `from 2 mon jan`
F	Starts the request on the first day of the specified time unit (weekend, business week, week number, month number). For example, to start a request on the first Monday in November, use these FROM options: `from f mon nov`
L	Starts the request on the last day of the specified time unit (weekend, business week, week number, month number). For example, to start a request on the last Monday in November, use these FROM options: `from l mon nov`
dayname	Starts a request on a specific day of the week (MON, TUE, WED, THU, FRI, SAT, or SUN)
WE	Starts a request on a weekend day
W	Starts a request at the beginning (Monday) of the specified week. For example, to start a request the beginning of the third week in July, use these FROM options: `from 3 w jul`

Option	Meaning
B	Starts a request on a business day
weeknumber	Starts a request the nth week from the current week: 001 means next week, 002 means the week after that, and so on. For example, to start a request on Tuesday of next week if specified on Monday of the current week, use the following FROM options: `from 1 tue 001`
M	Starts a request on the specified day or week of the current month. For example, to start a request on the third week of this month, use these FROM options: `from 3 w m`
monthname	Starts a request on the specified day or week of the named month (such as JAN or FEB). For example, to start a request on the second weekend day in January, use these FROM options: `from 2 we jan`
Q	Starts a request on the specified day or week of the current quarter. For example, to start a request on the last business day of this quarter, use these FROM options: `from l b q`
*n*Q	Starts a request on the specified day or week of the nth quarter. For example, to start a request the first week of the fourth quarter, use these FROM options: `from f w 4q`

ON *day*

Specifies the day of the week on which the EXEC is scheduled to execute. Valid day values are MON, TUE, WED, THU, FRI, SAT, and SUN. The initial run date is calculated to fall on the next occurrence of *day*. EXECs scheduled for a specific day cannot be repeated using the AGAIN DAILY or AGAIN MONTHEND options; however, any other AGAIN scheduling intervals can be specified. You can use ON with the FOR option as a repeat factor.

RELEASE *userid*

Specifies this request as a dependent request. Requests scheduled with the RELEASE option are never explicitly scheduled for execution. Therefore, the server ignores all scheduling parameters except WITHIN. The user ID specified is a user ID that is allowed to release the request for initiation using the RELEASE command. Other user IDs can release the request, including the user ID of the person owning the request and a user ID having NOPASS authority. Also, a person knowing the password of the person owning the request can release the request.

WITHIN {*hh*:*mm*:*ss* | *}

Specifies the amount of time *CA VM:Schedule* is to keep trying to start the request if it cannot start at the scheduled time. If the EXEC does not start within this amount of time, it is skipped. Your site can define the default initiation time limit. If you specify *, the WITHIN time is unlimited.

Repeat Options

[AGAIN [BI] *interval*]

Specifies how often you want to run the request.

BI repeats the request every other interval. For example, AGAIN BI DAILY repeats a request every other day. BI is not valid with the HOURLY, YEARLY, *number* DAYS, or *number* BDAYS options.

The following table describes the intervals and purpose:

Interval	Purpose
HOURLY	Starts the request every hour (not valid with BI)
DAILY	Starts the request every day; for example, AGAIN DAILY starts a request every day; AGAIN [BI] DAILY starts a request every other day
BDAILY	Starts the request every business day
WEEKLY	Starts the request every week
WEEKEND	Starts the requests every weekend day
MONTHLY	Starts the request every month
QTRLY	Starts the request every quarter
YEARLY	Starts the request every year (not valid with BI)
MONTHEND	Starts the request on the last day of the month
QTREND	Starts the request on the last day of the quarter
number DAYS	Starts the request every number of days (not valid with BI)

Interval	Purpose
number BDAYS	Starts the request every number of business days (not valid with BI)

DURING *options*

Repeats the request on the requested day or days during the specified period. Options are:

nnn

Repeats a request on the requested days every *nnn*th week (Monday through Sunday). For example, again daily during three schedules a request to run daily for a week every three weeks.

M

Repeats a request on the specified days or weeks of the current month; for example, on mon during m repeats a request every Monday during this month.

monthname

Repeats a request on the specified days or weeks of the named month (such as JAN or FEB). For example, again daily during feb repeats a request every day during February.

Q

Repeats a request on the specified days, weeks, or months of the current quarter; for example, again weekly during q repeats a request every week during this quarter.

nQ

Repeats a request on the requested days, weeks, or months of the specified quarter; for example, again daily during 1q repeats a request every day during the first quarter.

[ON *day*[-*day*]]...

Specifies the day of the week on which the request is scheduled to execute. Valid day values are MON, TUE, WED, THU, FRI, SAT, and SUN. The initial run date is calculated to fall on the next specified day. When you use this option with the DURING option, the server repeats the request on the requested day or days during the specified period.

Requests scheduled for a specific day cannot be repeated using the AGAIN BDAILY, AGAIN DAILY, AGAIN MONTHEND, AGAIN WEEKEND, or AGAIN QTREND options; however, any other AGAIN scheduling intervals can be specified. You can use ON with FOR option as a repeat factor.

You can specify the ON option multiple times, for example, on mon on wed on fri.

You can use a dash to specify a range of days for example, mon-fri, sat-sun.

EVERY *hh*:*mm*:*ss*

Specifies the interval of time until the next initiation of a request on a given day. The maximum interval is 23:59:59 and can extend beyond midnight.

Resource Limit Options

CPU *seconds*

Specifies the threshold for the total (virtual plus overhead) CPU in seconds that the autologged virtual machine can consume. If this value is exceeded, the server calls the MONITOR user exit. The CPU option is effective only if your site has implemented a MONITOR user exit routine to monitor resources used by the server requests.

SIO *count*

Specifies the threshold for the disk and tape I/Os the autologged virtual machine can perform during request execution. If this value is exceeded, the server calls the MONITOR user exit. The SIO option is effective only if your site has implemented a MONITOR user exit routine.

UIO *count*

Specifies the threshold for the unit record (reader, printer, punch) I/Os the virtual machine can perform during request execution. If this value is exceeded, the server calls the MONITOR user exit. The UIO option is effective only if your site has implemented a MONITOR user exit routine.

STORAGE *size*

Specifies the storage size in kilobyte (K) or megabyte (M) units of the virtual machine that executes the request. The specified size must be equal to or less than the virtual machine's maximum storage size as currently set in the CP directory.

Initiation Limits Options

FOR *number*

Specifies the number of times that you want to run the request. Requests scheduled with the FOR option must also include an ON, AGAIN, or EVERY option.

TOTIME *hh*:*mm*:*ss*

Specifies the latest time of day at which a request scheduled to repeat with the EVERY option is to initiate.

UNTIL *mm/dd/yy*

Specifies a date that limits the automatic rescheduling of the request. The request will run up to and including the UNTIL date.

All dates that you specify with the UNTIL option must contain a month and a day. If you omit the year, the default is the current year. The latest date that you can specify is December 31, 2041. *CA VM:Schedule* interprets the years 00-41 as 2000-2041.

UNTIL *options*

Runs a request up to, and including a given day, or week of a particular week, month, or quarter. The following table presents the UNTIL options:

Option	Purpose
dayname	Runs the request until the end of the named day (MON, TUE, WED, THU, FRI, SAT, and SUN). For example, to run a request through the end of the first Monday in November use these UNTIL options: `until f mon nov`
WE	Runs the request until the end of the specified weekend day
W	Runs the request until the beginning of the specified week (through Monday). For example, to run a request through the end of Monday of the last week in July, use these UNTIL options: `until l w jul`
B	Runs the request until the end of the specified business day
weeknumber	Runs the request until the nnnth week (3-digit weeks) from this week. For example, if on Monday you want to schedule a request to run daily up to and including Tuesday of the following week, use these UNTIL options: `until 1 tue 001`

Option	Purpose
M	Runs the request until the end of the specified day or week of the current month. For example, to run a request through the end of the third week of this month, use these UNTIL options: `until 3 w m`
monthname	Runs the request until the end of the specified day or week of the named month (such as JAN or FEB). For example, to run a request through the second weekend day in January, use these UNTIL options: `until 2 we jan`
Q	Runs the request until the end of the specified day or week of the current quarter. For example, to run a request through the end of the last business day of this quarter, use these UNTIL options: `until l b q`
nQ	Runs the request until the end of the specified day or week of the nth quarter. For example, to run a request through the end of Monday of the last week of the fourth quarter, use these UNTIL options: `until l w 4q`

Shift and Range Options

INSIDE {*shift* | *range*}

Specifies a name associated with a time or date period that the *CA VM:Schedule* system administrator defined. (Use the CONFIG command to determine the shifts and ranges that your site defined.) The request does not run unless the time specified with the AT or FROMTIME option falls inside the specified shift or range. The INSIDE option cannot be specified with the OUTSIDE option.

If you specify an INSIDE shift or range without using an AT time or FROM date, *CA VM:Schedule* uses as the default the first available time and day within the specified shift or range.

OUTSIDE {*shift* | *range*}

Specifies a name associated with a date or time period that the *CA VM:Schedule* system administrator defined. (Use the CONFIG command to determine the shifts and ranges defined at your site.) The request does not run unless the time specified with the AT or FROMTIME option falls outside the specified shift or range. The OUTSIDE option cannot be specified with the INSIDE option.

If you specify an OUTSIDE shift or range without using an AT time or FROM date, *CA VM:Schedule* uses as the default the first available time and day not included in the specified shift or range.

Log File Options

EXCEPT [SPOOL *userid*]

Specifies that only exceptional events (such as skipping a request, or request execution errors) should be recorded in the message file sent to the requesting user ID after *CA VM:Schedule* initiates a request. This is the default.

SPOOL *user ID* specifies the user ID of the virtual machine that is to receive messages. The default user ID is the user ID of the virtual machine scheduling the request. The SPOOL option is ignored if you also specify the NOLOG option.

LOG [SPOOL *userid*]

Specifies that a message file is to be sent to the virtual machine specified by the SPOOL option whenever the request initiates. (If the SPOOL option is not specified, the message file is sent to the virtual machine that scheduled the request.) You need special authorization to use the LOG option.

SPOOL *userid* specifies the user ID of the virtual machine that is to receive messages. The default user ID is the user ID of the virtual machine scheduling the request. The SPOOL option is ignored if you also specify the NOLOG option.

NOLOG

Specifies that a message file is not produced. You need special authorization to use the NOLOG option.

Monitoring Options

MONITOR {YES | NO}

Specifies whether request monitoring is to be performed. *CA VM:Schedule* uses configuration file settings (or system defaults) if you specify only some or no other monitoring options. If not specified in the VMSCHED CONFIG file, the default is NO. If MONITOR YES is specified in the VMSCHED CONFIG file, you cannot use the MONITOR NO option.

LOGOff {YES | NO}

Specifies whether to log off the requesting user ID when the request ends. Specifying LOGOFF YES automatically invokes request monitoring. If you specify NO, the user ID remains logged on. If not specified in the VMSCHED CONFIG file, the default is NO.

CONSOLE {YES | NO}

Spools the requesting user ID's console. If not specified in the VMSCHED CONFIG file, the default is NO. Specifying this option automatically invokes request monitoring.

CONSTO *userid*

Specifies the user ID to spool the console to. The default is the requesting user ID. Specifying the CONSTO option automatically invokes request monitoring.

DUMP {CP | VM | N}

Performs a CP or VM dump if the request abends. N specifies that no dump is to be generated. Specifying DUMP CP or DUMP VM automatically invokes request monitoring. If not specified in the VMSCHED CONFIG file, the default is N.

DUMPTO *userid*

Specifies the user ID to receive the dump. The default is the requesting user ID. The DUMPTO option cannot be used with DUMP N. Specifying the DUMPTO option automatically invokes request monitoring. If the type of dump is not specified by the user or in the VMSCHED CONFIG file, DUMP VM is assumed.

Miscellaneous Options

PASSWORD *password*

Specifies the CP LOGON password of the virtual machine being scheduled. If you do not specify a password or specify a **?**, *CA VM:Schedule* prompts for the password. If the SCHEDULE command is invoked by means of the CP SMSG command, you must specify the password explicitly on the command line.

Your *CA VM:Schedule* system administrator can specify that users not be allowed to enter the password in the options string, but instead must be prompted for the password, in which case CP SMSG cannot be used. *CA VM:Schedule* verifies your CP LOGON password when you issue the SCHEDULE command. You can change your password without affecting future initiations of the request.

SYSNAME {*systemname* | *}

Specifies the system name of the Single System Image member where this request is to run. The system name specified must be a member of the SSI cluster where *CA VM:Schedule* runs. Specify * if you want to clear this field in a request that has it already set. If not specified the request is run on the SSI member where *CA VM:Schedule* is running. This option is invalid if *CA VM:Schedule* is not configured for Single System Image mode.

USER *userid*

Specifies the user ID on which the request is to run. If not specified, the user ID defaults to the user ID that issued the SCHEDULE command. When you schedule a request to run on another user ID, make sure the other user ID either owns the scheduled program or EXEC, or has access to it.

Description

The SCHEDULE command schedules a program, EXEC, or CP or CMS command for initiation on an autologged virtual machine.

Make sure that your PROFILE EXECs and scheduled EXECs contain a SET BLIP OFF command to keep any resulting console spool files free from extraneous data.

Examples

- To run the REPORT request every business day, enter the following command:

  ```
  vmsched sched report report (again bdaily
  ```

- To run it every other day, enter the following command:

  ```
  vmsched sched report report (again bi daily
  ```

- To run it every other week, enter the following command:

  ```
  vmsched sched report report (again bi weekly
  ```

- To run it every other Wednesday, enter the following command:

  ```
  vmsched sched report report (on wed again bi weekly
  ```

- To run the report on the first business day of every month, enter the following command:

  ```
  vmsched sched report report (from f b monthname again monthly
  ```

- To schedule the REPORT command to run daily starting on the first business day of the third quarter, enter the following command:

  ```
  vmsched sched report report (again daily from f b 3q
  ```

- To schedule the CLEANUP command to run from today through the last business day of December, enter the following command:

  ```
  vmsched sched cleanup cleanup (again daily until l b dec
  ```

- To schedule your monthly backups to run on the first Saturday of every month at 11 P.M., enter the following command:

  ```
  vmsched sched mbackup mbackup (at 23:00 from f sat m again monthly
  ```

- To schedule the DEPT request to run on the first business day of January and again every other week until the last business day of the fourth quarter, enter the following command:

  ```
  vmsched sched dept dept (again bi weekly from 1 b jan until l b 4q
  ```

- To repeat the REPORT request every Monday of the current quarter, enter the following command:

  ```
  vmsched sched report report (on mon during q
  ```

- To schedule the CHECKUP request to run through January, beginning January 10, enter the following command:

  ```
  vmsched sched checkup checkup (again daily during jan from 01/10/yy
  ```

- To schedule the ACCTREP1 request to run every Friday at 18:00 on system SSI00123 enter the following command:

  ```
  vmsched sched acctrep1 (at 18:00 on fri again weekly sysname
  ssi00123
  ```

Duplicate Options

If you specify duplicate options on the command line, *CA VM:Schedule* uses the last option on the line; all previous option entries are ignored.

Note: This behavior does not apply to the ON option.

Start Times

If you do not specify a start time by using the AT or FROMTIME option, the start time defaults to a site-specified delay interval from the moment the command is ISSUED, not completed. If your site uses a short delay interval, the server might skip the first initiation of a request, if you take a long time to reply to prompts for command parameters or passwords.

The delay interval applies only to requests scheduled to run today. For example, suppose your system delay time is 5 minutes. If at 11 a.m. you schedule a request to run tomorrow without specifying a start time, the server schedules the request to run at 11 a.m. tomorrow, not 11:05 tomorrow. To find the default interval used at your site, use the CONFIG command.

Resource Limits

If your site implemented a MONITOR user exit routine to monitor the resources that your server requests use, the resources that you specify as options in the CHANGE or SCHEDULE command are monitored. For details about resource monitoring, see your *CA VM:Schedule* system administrator.

The CONFIG command displays the default values for CPU, SIO, and UIO, but only if the server monitors resource consumption for that parameter.

Your site can specify resource limits for classed requests. You can display these limits with the CONFIG command, even if a MONITOR user exit has not been implemented at your site. However, in this case resources are not monitored.

The resource limits for requests can be set in three locations:

- The USEREXIT MONITOR record. These limits (if set) are the default resource limits.
- The CLASS record (classed requests only). These limits are the maximum resource limits for the request processing class. These limits override any maximum limits that are set on the USEREXIT MONITOR record.
- Options on the SCHEDULE command. Limits that are set here override any maximum limits that are set on the USEREXIT MONITOR record. For classed requests, limits that are set here cannot exceed the limits (if any) set in the CLASS record. However, they can be more restrictive than those limits.

Request Logging

The server sends a spool file to the user ID scheduling a request when a request is initiated, unless you specify the NOLOG option. When you schedule a request with the FOR option, only one spool file is sent for the last successful initiation of the request.

If you do not specify either the LOG or the NOLOG option, the server sends a message file only when exceptional conditions occur. Exceptional conditions include the skipping of a request initiation due to unsuccessful autolog attempts, the final run of a request, and the occurrence of any type of request execution error. To determine if you are authorized to use the LOG or NOLOG option, check with your *CA VM:Schedule* system administrator.

When Requests Fail

If the virtual machine is logged on or another autolog error occurs, requests fail to initiate. If the failure occurs because the virtual machine is logged on, the initiation must begin within the WITHIN period, or the request is skipped. If the failure is because of any other autolog error, the request is canceled.

SET DISPLAY Command

The SET DISPLAY command lets you change your screen expertise level. After you change levels, the new level remains in effect until you change levels again.

```
SET Display {Novice | Fluent | Expert}
```

If you define a request at one level, and you later copy or update it while using a lower level, you may lose some parts of the request. In particular, you may lose repeat options available only at higher levels, request monitoring, logging, and resource allocation options. For example, suppose you define a request at the EXPERT level to repeat every other business day of the current quarter, and later copy it while using the FLUENT level. Because you cannot repeat requests this way from the FLUENT level, the copied request does not specify any repeat initiations. The request uses the default repeat setting, which is *not* to repeat the request.

Note: For more information about the different fluency levels, and about using CA VM:Schedule screens, see *Using Screens* (see page 19).

SKIP Command

The SKIP command lets you skip one or more subsequent initiations of a request.

```
SKip requestname [number] [([User userid][Password password]]
```

Definitions

requestname

Specifies the name of the request that you want to skip.

number

Specifies the number of initiations that you want to skip.

Default: 1

USER *userid*

Specifies the user ID of the virtual machine whose request you want to skip. This parameter is required to skip initiations of requests that are scheduled to execute on a user ID other than your own.

PASSWORD *password*

Specifies the CP logon password for the virtual machine whose request you want to skip. Your *CA VM:Schedule* system administrator can specify that users cannot enter the password in the options string, but instead must be prompted for passwords. If you use prompting, you cannot use CP SMSG to issue the command.

Description

The SKIP command skips one or more subsequent initiations of a request. The SKIP command is not valid for requests that are scheduled for only a single initiation. Use the DELAY or CANCEL commands to avoid the initiation of these requests.

You cannot use the SKIP command to skip the initiation of requests scheduled with the RELEASE option.

You can determine the next scheduled execution date and time with the QUERY command. The QUERY command displays the next actual initiation time after accounting for the skipped initiations.

If either of these conditions occur, the server cancels a request automatically:

- The number of initiations skipped causes the scheduled date to exceed the maximum rescheduling date you specified with the SCHEDULE command UNTIL option.
- The number of initiations skipped exceeds the remaining initiation limit that you specified with the SCHEDULE command FOR option.

Example

To skip the next two initiations of request PROD01, enter the following command:

```
vmsched skip prod01 2
```

TRANSFER Command

The TRANSFER command lets you change the user ID that runs a request.

```
TRANSfer {requestname | *} olduser newuser
```

Definitions

***requestname* | ***

Specifies the name of the request to be transferred to another user ID. Specifying * indicates that all requests are to be transferred.

olduser

Specifies the user ID on which the request is currently scheduled to run.

newuser

Specifies the user ID on which you want the request to run.

Authorizations

You must have TRANSFER authorization to use this command.

Description

The TRANSFER command lets you change the user ID that runs a request.

CA VM:Schedule checks for duplicate request names. You cannot transfer a request to another user who owns one with the same name. If you attempt to transfer a request that would result in a duplicate name, *CA VM:Schedule* rejects the transfer with an error message and a CMS return code of 24.

Examples

- ROGER has left the company. To make his supervisor LOUISE the owner of all his requests, enter the following command:

  ```
  vmsched transfer * roger louise
  ```

- Another user, ALICE, has been promoted. She will keep most of her requests, but one, ACCOUNTS, should now belong to LOUISE. To transfer that request to LOUISE, enter the following command:

  ```
  vmsched transfer accounts alice louise
  ```

WHEN Command

The WHEN command lets you list the times that a request runs. If you specify no options, the server displays up to 999 scheduled initiations for today.

```
When requestname [( "Options"]


Options:

["Selection options"]
["Output options"]
["Miscellaneous options"]


Selection options:

[FRom mm/dd/yy]
[NExt number]
[UNtil mm/dd/yy]

Output options:

[File filename] [PRint] [STack {FIFO | LIFO}] [Term]


Miscellaneous options:

[Password password]
[SOrt]
[User userid]
```

Definitions

requestname

Specifies the name of a request for which you want to list the run times. Use an asterisk to display all request names, or a trailing asterisk to list requests with similar names.

FROM *mm/dd/yy*

Specifies the date from which you want to list the run times for the request. The default is the current date.

NEXT *number*

Specifies how many initiations to list. The default is 999. The number of requests displayed is limited by the date range you specify. See *Controlling the Number of Runs Displayed* **(see page 136)** for more information about using this option.

UNTIL *mm/dd/yy*

Specifies the last date for which requests are to be displayed. If FROM is specified without UNTIL, the UNTIL date defaults to the FROM date.

FILE *filename*

Writes the output of the WHEN command to the file *filename* VMSCHED A1. If you do not specify the FILE option, the output is displayed on your terminal.

PRINT

Sends the output from the WHEN command to your virtual printer. If the PRINT option is not specified, the output displays on your terminal.

STACK { FIFO | LIFO }

Specifies that output is to be stacked first in, first out (FIFO), or last in, first out (LIFO) in your program stack.

TERM

Displays the output on your terminal. TERM is the default if no other output option is specified.

USER *userid*

Specifies the user ID whose requests' run times are to be listed. The default is the user ID that issued the WHEN command. If you specify another user ID, the CP logon password must be specified unless you have NOPASS authorization.

PASSWORD *password*

Specifies the CP logon password for the user ID whose requests are being queried. The password must be specified when you are listing run times for another user ID's requests. Your *CA VM:Schedule* system administrator can specify that users are not allowed to enter the password in the option string; instead, you are required to use the prompting procedure for password entry. You do not have to supply your password when listing times for requests you scheduled on your own user ID.

SORT

Sorts the list of scheduled requests by date and time. This option is useful when using pattern matching to display multiple request names. By default, repeat initiations within a given request name are already shown in order of date and time.

Description

The WHEN command lists the times that a request runs. If you specify no options, *CA VM:Schedule* displays up to 999 scheduled initiations for today.

Examples

- To display all requests that you scheduled to run on your user ID from September 7 through September 11, enter the following command:

  ```
  vmsched when * (from 09/07/yy until 09/11/yy
  ```

- You scheduled the CHECKUP request to run every 15 minutes today. To make sure the request was scheduled correctly, enter the following command to display the next 10 runs:

  ```
  vmsched when checkup (next 10
  ```

Output Destinations

You can specify the FILE, PRINT, STACK, and TERM options together on a single WHEN command line. Only the TERM option can be specified when entering the WHEN command from the *CA VM:Schedule* service virtual machine console. If none of the FILE, PRINT, STACK, or TERM options are specified, the output displays on your terminal.

Controlling the Number of Runs Displayed

If you leave out NEXT, FROM, and UNTIL, *CA VM:Schedule* lists the times and dates for up to the next 999 runs for today. If you specify just dates, *CA VM:Schedule* lists up to 999 runs for the date range specified.

Listing Many Requests

To avoid tying up your terminal while listing many requests, schedule the WHEN command to run later when you do not need your user ID, and print the output. For example, you want to display all requests scheduled to run on June 27 and 28. You place the following record in an EXEC, and you schedule the EXEC to run later:

```
vmsched when * (from 06/27/yy until 06/28/yy print sort
```

Index