SETUP Manager's Guide

February, 1990

This manual describes SETUP, a utility for centralizing application setup operations on VMS systems.

Revision/Update Information:	This manual supersedes the *SETUP User's Guide*.
Operating System and Version:	VMS V5.0 or later
Software Version:	SETUP V2.1

Engineering Computing Services
Rensselaer Polytechnic Institute
Troy, New York

27 February 1990

Permission is granted to copy and redistribute this document provide the copies are made for no commercial advantage.

The information in this document is subject to change without notice and should not be construed as a commitment by Rensselaer Polytechnic Institute. Rensselaer assumes no responsibility for any errors that may appear in this document.

DISCLAIMER: The software described in this document is provided "**as is**". No guarantee is made by the author or the author's employer as to the suitability, reliability, security, usefulness, or performance of this software.

Copyright ©1988, 1990 Rensselaer Polytechnic Institute

All Rights Reserved.
Printed in U.S.A.

This document was prepared using VAX DOCUMENT, Version 1.2

Preface

If you run many third-party applications on your VMS system, you probably have had to contend with the problems of getting these applications ‘‘set up’’ for users: defining the right logical names, symbols, commands, etc. When many applications are available, placing all of their definitions in the system-wide login command procedure can drastically slow down login time, and forcing users to place the appropriate commands in their own LOGIN.COM files can make for headaches when the commands required to set up an application change.

SETUP assists you with this task by providing one command through which all application setup can be executed. You define the application names and the command or commands to be executed to perform the setup. You can change the actual commands at any time without the users having to modify their LOGIN.COM files.

SETUP also allows you to restrict access to setup information and commands in two ways: by VAXcluster node name, and through the use of access control lists (ACL’s).

Intended Audience

This manual is intended for system managers responsible for installing and setting up SETUP.

Document Structure

This document consists of two parts. The first describes the installation and use of SETUP. The second describes the SETUP command and the SETUPDEF APPLICATION command.

New and Changed Features

SETUP V2.1 includes the following new features and changes.

1. The APPLICATION command in SETUPDEF now includes a /NODE qualifier to allow access restrictions based on node names.
2. Messages were improved in SETUPDEF.

Part I Management Guide

This part describes the installation and use of SETUP.

1 Installing SETUP

1.1 Installing the SETUP Files

The SETUP binary distribution includes the following files:

SETUP.EXE	The SETUP image
SETUPDEF.EXE	Program to create the SETUP application table
SETUP_CLD.CLD	Command Language Definition file for SETUP
SETUP.RNH	DSR source for SETUP help information

For a typical installation, you would perform the following steps:

```
$ COPY SETUP.EXE SYS$COMMON:[SYSEXE]/PROTECTION=W:RE
$ SET COMMAND/TABLE=SYS$COMMON:[SYSLIB]DCLTABLES-
_$   /OUTPUT=SYS$COMMON:[SYSLIB]DCLTABLES SETUP_CLD
$ RUNOFF SETUP.RNH
$ LIBRARY/REPLACE/HELP SYS$HELP:HELPLIB SETUP
```

You should then use the INSTALL utility to replace SYS$LIBRARY:DCLTABLES the system, and if the system is part of a VAXcluster, the other nodes in the cluster. If you plan on using the security features of SETUP, you should also use the following command:

```
$ INSTALL CREATE SYS$SYSTEM:SETUP/PRIV=SYSPRV
```

You should also include this command in you system startup command procedure.

1.2 Creating the Application Table

The SETUPDEF program is used to build the application table used by SETUP. By default, SETUPDEF takes its input from the file SETUP_CMDS.DAT in the current directory, and writes the table to the file RPI$ROOT:[SETUP]SETUP_SECTION.DAT. However, by defining the logical names SETUP_CMDS and SETUP_SECTION, you can redirect the input and/or output of the program.

For a typical installation, it is best to create a directory for storing SETUPDEF, any SETUP_CMDS.DAT files, and the SETUP_SECTION.DAT file. Unless this directory is called RPI$ROOT:[SETUP], you should define the following system-wide logical name to point to the application table file:

```
$ DEFINE/SYSTEM/EXEC SETUP_SECTION dev:[dir]SETUP_SECTION
```

This command should also be included in your system startup command procedure, and should be executed on all nodes in your cluster if you run a VAXcluster system.

1.2.1 Defining Applications

To ease the definition of SETUP applications, create a SETUP_CMDS.DAT file to store the APPLICATION commands used by SETUPDEF. The APPLICATION command is described in Part II of this document.

It is important to assign meaningful but brief names to your applications. If you maintain multiple versions of some applications simultaneously, you may want to include some kind of version information in the name. For example, when a new version of an application comes in, you could define it as application NEW_*name*. When it has been tested, you can then make the new version available as just *name* while moving the older version to OLD_*name*. This way, when users just include SETUP *name* in their LOGIN files, they get the most current version considered stable.

1.2.2 Ordering the Definitions

The SETUPDEF program stores the definitions in the application table in the order that they are given, without any form of sorting. You may wish to sort the definitions in your SETUP_CMDS.DAT file so that they are either in alphabetic order for better listings, or in most-used-first order for faster access (though access should be fast enough unless the application table is inordinately large).

1.2.3 Running SETUPDEF

Once you have entered your APPLICATION commands into the SETUP_CMDS.DAT file, just type RUN SETUPDEF to create the new application table. SETUPDEF will display an informational message about each application it places in the table. If SETUP is installed with SYSPRV, no further action is needed; otherwise, you should be sure that the SETUP_SECTION file is protected WORLD:READ.

1.3 Using Access Restrctions

SETUP provides two forms of access restrictions: node-based restrictions and access control lists (ACL's). Both can be used to tailor a single application table for use by several groups of users.

Node-based restrictions are useful when you have an application that is licensed only on certain nodes in your VAXcluster, or for any other reason you may have for restricting its use to certain nodes. The application shows up in SETUP/LIST on all nodes, but an attempt to use SETUP on an unauthorized node returns the message ‘‘application not available on this node’’. Users can type SETUP/LIST/FULL to display the list of nodes on which the application may be used.

ACL's are useful when you have a centrally managed application that should be used only by a certain group of users. You can key the ACL's by UIC or on an identifier. READ (which allows SETUP/LIST access) and/or EXECUTE (which allows the execution of the setup) can be granted.

If ACL's are used to protect the setup information, you should also be sure that the application table, the setup command procedures, and applications themselves are adequately protected. Generally, users will need only EXECUTE access to setup command procedures and the directories in which they are stored. No WORLD access to the SETUP_SECTION file is required if SETUP is installed with SYSPRV or READALL.

1.4 Building Robust Setup Procedures

Most applications will need to have command procedures executed for their setups. You should be sure that these command procedures are coded so that

- they define what they need without interfering with other symbols, logical names, etc. the user may need;
- they cannot be tripped up by unusual symbols, logicals, etc., the user may have defined;
- they can be executed multiple times without any ill effects.

Some third-party application vendors don't bother to code the setup command procedures they provide so that they abide by these rules, so it is prudent to examine any provided procedures carefully and modify them as needed.

The following is an example of the kind of command procedure that follows the robustness rules listed above.

```
$ V = 'F$VERIFY (0)                ! prevent SET VERIFY interference
$! Setup for ACSL
$!
$ SET := SET                       ! prevent symbol interference
$ IF := IF
$ DEFINE := DEFINE/NOLOG           ! use /NOLOG to eliminate "superseded" msgs
$ EXIT := EXIT
$!
$ SET NOON
$ DEFINE ACSL_SYS ECS_SYS6:[ACSL.ACSL8.S]
$ DEFINE ACSLLIB ACSL_SYS:ACSLL8R.OLB
$ DEFINE ACSLSHR ACSL_SYS:ACSLL8R.EXE
$ DEFINE ACSLHLP ACSL_SYS:ACSLH8R.HLB
$ DEFINE ACSLSPR ACSL_SYS:ACSLSPR.OLB
$ DEFINE ACSLKEY ACSL_SYS:ACSLKEY.DAT
$ DEFINE ACSLOPT ACSL_SYS:ACSL.OPT
$ SET COMMAND ACSL_SYS:ACSL
$!
$! The following DCL code adds the ACSL help library to the
$! process-local list of help libraries.  It first checks to
$! see if the library name is already present, and if not,
$! adds it to the search list logical name definition.
$!
$ HLP_NAME := ACSL_SYS:ACSLH8R
$ HLP = F$TRNLNM ("HLP$LIBRARY","LNM$PROCESS")
$ IF HLP .EQS. HLP_NAME THEN EXIT 1+0*F$VERIFY(V)
$ I = 0
$ IF HLP .NES. "" THEN GOTO HLP_LOOP
$ DEFINE HLP$LIBRARY 'HLP_NAME
$ EXIT 1+0*F$VERIFY(V)
$HLP_LOOP:
$ I = I + 1
$ IF I .GT. 9 THEN EXIT
```

```
$ HLP = F$TRNLNM ("HLP$LIBRARY_''I'","LNM$PROCESS")
$ IF HLP .EQS. HLP_NAME THEN EXIT 1+0*F$VERIFY(V)
$ IF HLP .NES. "" THEN GOTO HLP_LOOP
$ DEFINE HLP$LIBRARY_'I' 'HLP_NAME
$ EXIT 1+0*F$VERIFY(V)
```

1.5 Sample SETUP_CMDS File

The following is a sample SETUP_CMDS file which contains several application definitions.

```
!
! ABAQUS: can be used only by authorized users
!
APPLICATION ABAQUS/SETUP="@RPI$ROOT:[SETUP]ABAQUS7"-
    /DESC="ABAQUS V4.7"-
    /ACCESS=((IDENTIFIER=ABAQUS_USER,ACCESS=R+E),-
          (IDENTIFIER=[SYS,*],ACCESS=R+E))

APPLICATION ACSL/SETUP="@RPI$ROOT:[SETUP]ACSL9"-
    /DESCRIPTION="Advanced Continuous Simulation Language Version 9B"

APPLICATION DISSPLA/SETUP="@RPI$ROOT:[SETUP]DISSPLA"-
    /DESCRIPTION="CA-DISSPLA graphics package Version 10.5"

!
! MACSYMA: only available on the 6320's
!
APPLICATION MACSYMA/SETUP="@RPI$ROOT:[SETUP]MACSYMA"-
    /DESCRIPTION="MACSYMA 412.61"-
    /NODE=(ECS01,ECS02)

APPLICATION MOVIE.BYU/SETUP="@RPI$ROOT:[SETUP]MOVIE-BYU"-
    /DESCRIPTION="MOVIE.BYU V6.2"

APPLICATION NEW_DISSPLA/SETUP="@RPI$ROOT:[SETUP]NEW_DISSPLA"-
    /DESCRIPTION="CA-DISSPLA graphics package Version 11"

APPLCIATION TELL-A-GRAF/SETUP="@RPI$ROOT:[SETUP]TELL-A-GRAF"/DESC=-
    "CA-Tell-a-Graf graphics generator V6.0"

APPLICATION TEX/SETUP="@RPI$ROOT:[SETUP]TEX"/DESC="TEX V2.991"

!
! Applications with the UNSP: prefix are "unsupported".
!
APPLICATION UNSP:EMACS/SETUP="@RPI$ROOT:[SETUP]EMACS"-
    /DESC="GNU Emacs version 18.52 (UNSUPPORTED)"
```

Part II Command Descriptions

SETUP

Application setup command.

FORMAT

SETUP *appl-name*

Command Qualifiers	Defaults
/FULL	
/INITIAL	
/LIST	

PARAMETERS

appl-name

The name of the application to be set up. When the /LIST qualifier is used, this can be a pattern including wildcard characters.

DESCRIPTION

The SETUP command is used to provide the logical names, symbols, commands, etc. needed to use a particular application. Typically, each application needs to be set up for each terminal session or batch job, so it may be a good idea to include the SETUP command for the applications you use in your LOGIN.COM file.

Some applications may also require a one-time setup, which might create files or directories needed for use of the application, etc. Any applications requiring a one-time setup will be listed as such when you use the SETUP/LIST command.

QUALIFIERS

/FULL

Used only with the /LIST qualifier, /FULL specifies that a more informative listing of each application should be displayed, including a one-line description of the application (if available) and any per-node availability restrictions.

/INITIAL

Specifies that the one-time setup commands for the application should be executed, instead of the per-login setup commands. If there is no one-time setup for the application, an error message is displayed.

/LIST

Lists the available applications, including the application name and types of setups available (per-login and/or one-time).

APPLICATION

In the SETUPDEF utility, defines an application setup.

FORMAT

APPLICATION *appl-name*

Command Qualifiers	Defaults
/ACCESS=acl	
/DESCRIPTION=desc	
/INITIAL=onetime-cmd	
/NODE=node-list	
/SETUP=perlogin-cmd	

PARAMETERS

appl-name

Name of the application to be defined. The name may consist of any characters, but should probably not include spaces or tabs, asterisks (‘‘*’’), or percent signs (‘‘%’’).

QUALIFIERS

/ACCESS=acl

Specifies an access control list to govern access to the application setup information. The access control list should contain identifer ACE’s which may grant EXECUTE and/or READ access. EXECUTE access is required for a user to execute the application setup command; READ access is required for a user to see the application definition with SETUP/LIST.

If you do not use the /ACCESS qualifier, access is assumed to be granted to all users for any operation. If you do use the /ACCESS qualifier, access is assumed to be denied to all users not covered by the ACL.

Using /ACCESS to protect setup information is not effective unless the SETUP_SECTION file is also protected (with no WORLD access), which will require that SETUP be installed with SYSPRV or READALL privilege. You should also protect the setup command procedures themselves (giving WORLD only EXECUTE access) as well as the directory or directories in which they are stored (also W:E access).

/DESCRIPTION=desc

Attaches a brief description of the application to the definition for display with SETUP/LIST/FULL. The description should be enclosed in quotation marks and should probably included the complete name of the application and the version.

/INITIAL=onetime-cmd

Defines the one-time setup command for the application. Any command, including ‘‘@’’ for command procedure execution, is allowed. If omitted, the application is assumed to have no one-time setup.

/NODE=node-list

Specifies that the application is only available on VAXcluster nodes matching one of the node names specified in *node-list*. The specified node names may include wildcard characters; if more than one is specified, the list should be surrounded by parentheses and separated by commas. Attempts to use SETUP for the application on a non-matching node results in an informational message indicating that the application is not available.

If omitted, the application is assumed to be available on all nodes.

/SETUP=perlogin-cmd

Defines the per-login setup command for the application. Any command, including ''@'' for command procedure execution, is allowed. If omitted, the application is assumed to have no per-login setup.