

# **EView/390z Discovery for z/OS**

## **HP UCMDB Integration**

### **Installation Guide**

**Software Version: 6.4**



**July 2013**

Copyright 2013 EView Technology, Inc.

## Legal Notices

### Warranty

*EView Technology makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. EView Technology shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.*

### Restricted Rights Legend

All rights are reserved. No part of this document may be copied, reproduced, or translated to another language without the prior written consent of EView Technology, Inc. The information contained in this material is subject to change without notice.

Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 for DOD agencies, and subparagraphs (c) (1) and (c) (2) of the Commercial Computer Software Restricted Rights clause at FAR 52.227-19 for other agencies.

EView Technology, Inc.  
4909 Green Road  
Raleigh, North Carolina 27616  
United States of America

### Copyright Notices

Copyright 2013 EView Technology, Inc.

No part of this document may be copied, reproduced, or translated into another language without the prior written consent of EView Technology, Inc. The information contained in this material is subject to change without notice.

### Trademark Notices

EView/390® is a registered trademark of EView Technology, Inc.

S/390, OS/390, z/OS, and zSeries are trademarks of International Business Machines Corporation.

Microsoft®, Windows®, and Windows NT® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of the Open Group.

All other product names are the property of their respective trademark or service mark holders and are hereby acknowledged.

## Table of Contents

<b>Conventions</b> .....	<b>6</b>
Introduction .....	6
<b>Documentation Map</b> .....	<b>8</b>
EView/390 Discovery Printed Manuals .....	9
EView/390 Online Information .....	9
<b>Installing and De-installing EView/390z</b> .....	<b>11</b>
Installation Requirements .....	12
Hardware Requirements .....	12
Software Requirements .....	12
Obtaining License Keys .....	13
Installing EView/390z on the UCMDB Probe .....	14
Installation Steps .....	14
Installed File Locations on the Management Server .....	14
Installing EView/390z on the z/OS Systems .....	15
What to Upload .....	15
Transferring Files to the Mainframe .....	15
Extracting Partitioned Datasets from Sequential Datasets .....	16
De-installing EView/390z .....	17
To Remove EView/390z Components from the Discovery Server .....	17
To Remove EView/390z from the z/OS systems .....	17
<b>Updating Mainframe Software</b> .....	<b>19</b>
Phase 1: Updating TCP/IP Connectivity .....	21

Reserving Port Numbers in PROFILE.TCPIP .....	21
Identifying the active TCPIP.DATA file .....	21
Phase 2: Updating z/OS.....	21
Authorizing the <i>hlq</i> .LOAD Dataset .....	21
Setting the Performance Group or Service Class .....	22
Adding an Entry to the Program Properties Table.....	22
Adding an Entry to the RACF Class.....	23
Phase 3: Updating EView/390 Parameter Cards .....	24
CMD Parameter Card .....	24
DELAY Parameter Card.....	25
FILTER Parameter Card .....	25
NLS Parameter Card .....	26
OSINFO Parameter Card.....	27
PRINTCARDS Parameter Card .....	27
MQMODEL Parameter Card MQCOMMAND Parameter Card MQDYNAMIC Parameter Card .....	28
RESTART Parameter Card.....	28
TCP Parameter Card .....	29
<b>Security and Discovery Requirements.....</b>	<b>31</b>
CICS .....	33
DB2.....	33
z/OS Console Authority.....	34
IMS.....	34
WebSphere MQ .....	34
<b>Starting and Stopping the Mainframe Component.....</b>	<b>37</b>
Running EView/390z as a Started Task.....	38
To Start the VP390 Job as a Started Task.....	38
To Stop the VP390 Task.....	38
Running EView/390 as a Batch Job .....	38
To Start the VP390 Job as a Batch Job .....	38
To Stop the VP390 Batch Job.....	38

<b>RACF Resource Class Permissions .....</b>	<b>39</b>
Security Settings for z/OS Resources .....	40
Extended Console Definitions in RACF .....	42

# Conventions

## Introduction

The following typographical conventions are used in this manual.

**Table 1-1: Typographical Conventions**

Font	Meaning	Example
<i>Italic</i>	Book or manual titles, and man page names	See the <i>EView/390z Discovery for HP UCMDB Installation Guide</i> for more information.
	Provides emphasis	You <i>must</i> follow these steps.
	Specifies a variable that you must supply when entering a command	At the prompt, enter rlogin <i>your_name</i> where you supply your login name.
	Parameters to a function	The <i>oper_name</i> parameter returns an integer response.
<b>Bold</b>	New terms	The monitor <b>agent</b> observes...
Computer	Text and items on the computer screen	The system replies: Press Enter
	Command names	Use the <code>dir</code> command ...
	Function names	Use the <code>opc_connect()</code> function to connect...
	File and directory names	<code>C:\windows\</code>
	Process names	Check to see if <code>opcmona</code> is running.
	Window/dialog box names	In the Add Logfile window...
<b>Computer Bold</b>	Text that you must enter	At the prompt, enter <b>dir</b>
<b>Keycap</b>	Keyboard keys	Press <b>Return</b> .

Font	Meaning	Example
[Button]	Buttons on the user interface.	Click [Operator] . Click the [Apply] button.
Menu Items	A menu name followed by a colon ( : ) means that you select the menu, then the item. When the item is followed by an arrow ( -> ), a cascading menu follows.	Select Actions:Utilities ->Reports ...

## Documentation Map

EView/390z Discovery for z/OS (EView/390 Discovery) provides a set of manuals that help you use the product and understand the concepts underlying the product. This section describes what information is available and where you can find it.



In addition to EView/390 documentation, related HP UCMDB products provide a comprehensive set of manuals that help you use the products and improve your understanding of the underlying UCMDB concepts.



# EView/390 Discovery Printed Manuals

This section provides an overview of the printed manuals and their contents.

## *EView/390z Discovery for z/OS Installation Guide*

Explains how to install, de-install, and configure EView/390 Discovery. Also includes how to upload installation files from the UCMDB probe, update EView/390, and start and stop EView/390 processes.

## *EView/390z Discovery for z/OS Administrator's Reference*

Explains how to customize and use EView/390. Also includes troubleshooting procedures and explanations of EView/390 system messages.

## EView/390 Online Information

The following information is available online:

- *EView/390 Discovery for z/OS Installation Guide*
- *EView/390 Discovery for z/OS Administrator's Reference*



## Installing and De-installing EView/390z

This chapter describes how to install and de-install EView/390z Discovery for z/OS (EView/390).

EView/390z Discovery for z/OS consists of two components. The “Client” component is installed on the HP UCMDB discovery probe where the mainframe discovery adapter is installed. The “Agent” component is installed on each z/OS operating system partition that will be discovered.

The EView/390z Discovery for z/OS is installed first on the DDMA probe and includes the Agent software files which are then transferred to the z/OS partitions for installation.

# Installation Requirements

This section describes the operating system, hardware, and software requirements for installing EView/390 software. To avoid problems during installation, read this section before you start the installation process.

## Hardware Requirements

- UCMDB Discovery Probe

EView/390 requires appropriate Ethernet hardware on the Discovery probe to communicate via TCP/IP.

All other hardware requirements are the same as the requirements for HP UCMDB.

- z/OS Operating System

EView/390 requires the appropriate Ethernet hardware on the S/390 to allow for TCP/IP communication with the UCMDB probe.

In addition, make sure that the Discovery probe and z/OS partitions meet the disk space requirements described in Table 3-1.

**Table 3-1: Additional Disk-Space Requirements**

Platform	Disk Space
UCMDB Discovery Probe	5MB
S/390 Mainframe	60 tracks of 3390 DASD

## Software Requirements

- On the Discovery Probe:
  - Microsoft Windows 2003 Server or higher.
  - UCMDB 8 or 9 with the Mainframe Adapter.
  - The TCP/IP network protocol stack must be active.
  - Active Perl version 5.8 or later from [www.activestate.com](http://www.activestate.com)

All other software requirements are the same as the requirements for HP UCMDB.

- On the z/OS operating system:
  - z/OS V1R3 or higher.
  - The TCP/IP network protocol stack (V3R1 or higher) must be active.

## Obtaining License Keys

EView/390z requires a license key to be applied to the configuration of each z/OS system that is to be managed by the Discovery probe. A license is required for each physical mainframe machine. The same license key is used for multiple LPARs on the same physical system. Contact EView Technology at +1-919-878-5199 or e-mail [support@eview-tech.com](mailto:support@eview-tech.com) to get the necessary license keys. Be prepared to give the serial number of the S/390 system (or first logical serial number in a multi-CPU LPAR system). The serial number can be found by issuing a `DISPLAY M=CPU` command from a z/OS console and reading the last five characters of the CPC ND line of the output.

For example, the following output shows that the serial number of this particular S/390 system is 70571:

### **DISPLAY M=CPU**

```
IEE174I 06.00.00 DISPLAY M
PROCESSOR STATUS
ID  CPU                SERIAL
0   +                0105717060
1   +                1105717060
2   +                2105717060
3   +                3105717060

CPC ND = 007060.H30.IBM.02.000000070571
CPC SI = 7060.H30.IBM.02.0000000000070571
CPC ID = 00
```

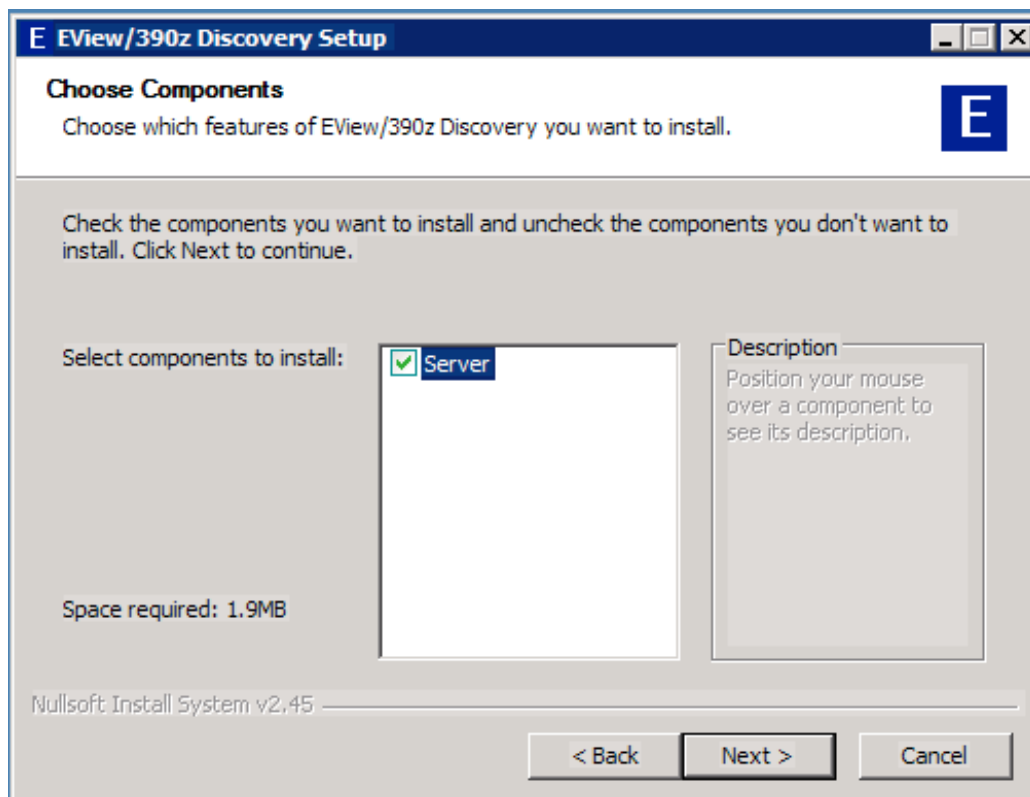
# Installing EView/390z on the UCMDB Probe

The EView/390z installation program is run as an executable on the UCMDB probe system.

## Installation Steps

1. Insert the EView/390z installation CD into the CD drive of UCMDB probe.
2. If the setup wizard does not automatically start, go to the top level directory on the installation CD and double-click `Eview390DiscInstall.exe`.

Figure 3-1: EView/390 Installation



## Installed File Locations on the Management Server

The installation process copies the necessary files to the UCMDB probe in the directory path you specified. The default path for EView/390 files is:

```
\Program Files\EView Technology\EView 390\
```

## Installing EView/390z on the z/OS Systems

To install the EView/390z Discovery agent on the managed nodes, use the File Transfer Protocol (FTP) to upload the EView/390z datasets to all z/OS mainframe LPARs that are to be discovered.

### What to Upload

After installing EView/390z on the UCMDB probe, the mainframe datasets are located in the following directory:

```
\Program Files\EView Technology\EView 390\mf
```

This directory contains the EView/390 files required for uploading to the z/OS mainframe, as shown in Table 3-2.

**Table 3-2: EView/390z Files to Upload to the S/390 Mainframe**

File Name	Description	3390 DASD Tracks
EV390.V64.LOAD.SEQ	EView/390 agent executables	45
EV390.V64.SAMP.SEQ	Sample JCL, startup parameter cards	5
EV390.V64.CLIST.SEQ	Command lists for executing mainframe commands	5

### Transferring Files to the Mainframe

Use FTP to send the several files from the UCMDB probe system to the z/OS system. Use binary mode when transmitting the files, and use the SITE or LOCSITE command to force ftp to create the target datasets with attributes DCB=(DSORG=PS,RECFM=FB,LRECL=80,BLKSIZE=3120) and a primary allocation of 45 DASD tracks.. Consult your mainframe systems programmer for the appropriate dataset high-level qualifier (hlq) name for the files as they are transferred to the S/390:

```
C:> cd Program Files\EView Technology\EView 390\mf
C:> ftp s390name
User: username
Password: ****
ftp> bin
ftp> quote site blksize=3120
ftp> quote site lrecl=80
ftp> quote site recfm=fb
ftp> quote site primary=45
ftp> put EV390.V64.LOAD.SEQ hlq.EV390.V64.LOAD.SEQ
ftp> put EV390.V64.SAMP.SEQ hlq.EV390.V64.SAMP.SEQ
ftp> put EV390.V64.CLIST.SEQ hlq.EV390.V64.CLIST.SEQ
ftp> quit
```

If you receive a B37 or D37 "out of space" error from any of the **put** commands, you may need to pre-allocate the sequential file on the mainframe using the sizes given in Table 3-2.

## Extracting Partitioned Datasets from Sequential Datasets

After uploading the files to sequential datasets on the mainframe, use the TSO RECEIVE command to extract a partitioned dataset (PDS) from each of the sequential datasets.

From a TSO command line, enter the following command for each of the uploaded datasets:

```
RECEIVE INDS ('hlq.dataset.SEQ')
```

The RECEIVE command will prompt you for additional restore parameters. The output PDS name can be modified at this time by entering the DA parameter. For example, to change the SAMP dataset HLQ to "EV390" enter the following:

```
DA ('EV390.V64.SAMP')
```



It is recommended that the EView/390z datasets contain the version number.



# De-installing EView/390z

This section describes how to remove EView/390z software from the following:

- UCMDB probe
- S/390 managed nodes

## To Remove EView/390z Components from the Discovery Server

Use the “Add/Remove Programs” utility from Windows Control Panel to remove EView/390z files and registry entries.

## To Remove EView/390z from the z/OS systems

To remove EView/390z from the managed nodes, follow these steps:

1. Stop the EView/390z task on the z/OS system.

To find out how to stop the EView/390 job on the managed nodes, see “*Running EView/390z as a Started Task*” in Chapter 6.

2. Delete the EView/390z datasets installed on the z/OS system.



## Updating Mainframe Software

This chapter contains instructions for updating z/OS resources on the mainframe, and updating the EView/390z input parameter cards to customize the mainframe task for the particular needs of your site.



# Phase 1: Updating TCP/IP Connectivity

The following modifications need to be made to the IBM TCP/IP:

- Reserve port numbers in `PROFILE.TCPIP`
- Identify TCP/IP high-level quantifier

## Reserving Port Numbers in PROFILE.TCPIP

Choose two available port numbers for use by EView/390 and identify them in the list of `PORT` values in the `PROFILE.TCPIP` dataset:

```
6106 TCP VP390
6107 TCP VP390
```

(The default ports used by EView/390 are 6106 and 6107. The default job name for EView/390 is "VP390".)



This step is optional. If specific port numbers are not reserved for EView/390 use, the EView/390z client connection will still succeed, but this reservation will flag the chosen port numbers for exclusive use by EView/390 so they are not used by other products on the mainframe.

## Identifying the active TCPIP.DATA file

Make note of the dataset/member location of the active `TCPIP.DATA` file for the TCP/IP stack that EView/390 will be connecting to. This dataset name will be needed in Chapter 6 when defining the `SYSTCPD DD` card in the startup JCL job.

# Phase 2: Updating z/OS

To run EView/390z, modify the z/OS datasets as follows:

- Authorize the `hlq.LOAD` dataset
- Set the performance group or add an entry to Workload Manager
- Add an entry to the Program Properties Table
- Add an entry to the RACF class

Inform the mainframe system programmer of changes needed to the `SYS1.PARMLIB` members.

## Authorizing the hlq.LOAD Dataset

Add the EView/390 `hlq.LOAD` dataset and its DASD volume name to the list of APF authorized datasets in one of the following:

- `SYS1.PARMLIB(IEAAPFxx)`

- `SYS1.PARMLIB (PROGxx)`

This addition is required to allow EView/390z to process certain authorized commands and perform security checks.

The authorization added to `SYS1.PARMLIB` takes effect after the next IPL. To dynamically authorize the `hlq.LOAD` dataset on DASD volume `volser` without an IPL, enter the following z/OS command:

```
SETPROG APF,ADD,DSNAME=hlq.LOAD,VOLUME=volser
```

## Setting the Performance Group or Service Class

If Workload Manager (WLM) is present on the S/390 system, add an entry for the VP390 job to the SYSTEM or SYSSTC service class, assigning it a priority slightly lower than VTAM.

If WLM is not used, set the performance group by adding a `TRXNAME` parameter for EView/390 to the STC subsystem definition of `SYS1.PARMLIB (IEAICSxx)`.

In the `TRXNAME` line, specify one of the following:

- Same performance group used by NetView/390 (if present)
- Performance group that is one level below the VTAM performance group

This addition ensures that EView/390z receives enough CPU time to avoid a backlog of network information processing. The default name for the EView/390 startup job is `VP390`.

For example, if NetView/390 is running in performance group 8, specify the addition for VP390 with the following:

```
TRXNAME=VP390,PGN=8
```

To dynamically reload the ICS file after a new entry is added, enter the following z/OS command:

```
SET ICS=xx
```

where `xx` is the two-digit suffix of the edited member.

## Adding an Entry to the Program Properties Table

Add a PPT entry to the `SYS1.PARMLIB (SCHEDxx)` for VP390, identifying the started task as a non-swappable, non-timed system task. This addition ensures that the VP390 address space is not swapped and that the job is not terminated when no network activity occurs.

The syntax for the PPT entry is as follow:

```
PPT PGMNAME (VP390)  
NOSWAP  
SYST
```

To dynamically reload the PPT after a new entry is added, enter the following z/OS command:

```
SET SCH=xx
```

where `xx` is the two-digit suffix of the edited member.

## Adding an Entry to the RACF Class

The VP390 task requires a z/OS UNIX System Services (USS) segment. Because USS segments are associated with RACF-defined user IDs, you should add an identifying entry for VP390 to a RACF class to meet the USS requirement.

This addition allows the VP390 to run as a started task. If the VP390 is to be run as a submitted job, enter the user ID on the `JOB` card of the startup job.

To add an entry to the RACF class, follow these steps:

1. Verify that the `STARTED` class is defined by entering the following command:

```
RLIST STARTED *
```

This command displays a list of entries for the `STARTED` class.

2. Determine whether a RACF user (for example, `IBMUSER`) has an OMVS segment by entering the following command:

```
LU IBMUSER OMVS
```

3. If the `STARTED` class is activated, add the VP390 task to the defined user (for example, `IBMUSER`) by entering the following:

```
RDEFINE STARTED VP390.VP390 STDATA(USER(IBMUSER) GROUP(SYS1))
```

Then refresh the class by entering the following:

```
SETROPTS RACLIST(STARTED) REFRESH
```

4. If the `STARTED` class is not activated, assign RACF identities to the started procedures.

Incorporate the following sample into the `ICHRIN03` job of `SYS1.SAMPLIB(RACTABLE)`

Example:

```
ICHRIN03 CSECT
COUNT   DC      AL2(((ENDRINO3-COUNT-2)/32)+32768)

*-----New VP390 Entry-----

ENTRY1   EQU      *
PROC1    DC        CL8'VP390   '
USERID   DC        CL8'IBMUSER '
GROUP1   DC        CL8'SYS1   '
FLAGS1   DC        XLI'00'
          DC        XL7'00'

*-----Last Entry-----

ENTRY2   EQU      *
PROC2    DC        CL8'*       '
USERID2  DC        CL8'IBMUSER '
GROUP2   DC        CL8'='     '
FLAG2    DC        XLI'00'
ENDRINO3 EQU      *
END
```

RACF allows the started procedures table to contain a generic entry, indicated by an asterisk (\*) in the procedure-name field. When searching the table for a procedure-name match, if RACF finds a procedure name of "\*" as the last entry in the table and the procedure name was not specifically matched by any other entry in the table, RACF uses the "\*" entry as a match for the procedure. This procedure is documented in *the IBM Security Server (RACF) System Programmer's Guide*.

## Phase 3: Updating EView/390 Parameter Cards

Modify initialization parameter cards to match the resources you have configured for the EView/390 client configuration. The parameter card dataset member(s) are pointed to by the SYSIN DD of the started task.

The EView/390 parameter cards are located in:

`hlq.SAMP (DDMPARM)`

Parameter cards may contain system symbols to ensure uniqueness if the same dataset member is being used for multiple LPARs. Symbols must start with an ampersand (&) and must end with a period if the symbol substitution occurs in the middle of a string. For example, to incorporate the value of &SYSNAME. into the name of the console defined for the CMD subtask, the parameter card would look like:

```
CMD &SYSNAME.CON
```

- Strings resulting from symbol substitution must conform to the length and syntax requirements of the parameter.

A detailed description of the parameter cards follows.

### CMD Parameter Card

Provides EView/390z with the ability to issue z/OS (MVS) commands.

#### Valid Values

```
consname [LOG|NOLOG] [HC={YES|NO}]
```

#### Sample Syntax

```
CMD EVOCONS2 NOLOG HC=YES
```

#### Description

This card will initialize the CMD subtask, which is used to send z/OS (MVS) commands from EView/390z to the mainframe OS.

#### Parameters



<i>consname</i>	Required. Specify a 1-8 character name for the extended MCS console you wish to define for issuing z/OS (MVS) commands. If this name is defined in RACF, the OPERPARM values in the RACF entry for this name are used for the console definition. Otherwise, a console is defined with default parameters AUTH=MASTER and ROUTCDE=NONE. (See Appendix A for information on defining an extended console to RACF.)
LOG   <u>NOLOG</u>	Optional. Specify LOG to force a system log message to be written for all z/OS commands entered from an EView/390 server. The commands are recorded in the system log with an EVO033 message. Specifying NOLOG here suppresses the writing of the EVO033 message. NOLOG is the default.
HC={ YES   <u>NO</u> }	Optional. Specify HC=YES to record all commands and responses from this extended console in the mainframe hardcopy log. HC=NO will prevent any hardcopy logging of the commands that are sent in from the EView/390 client.. HC=NO is the default

## DELAY Parameter Card

Number of seconds to wait until the next attempt to restart a subtask.

### Valid Values

1 to 86400 (seconds)

### Sample Syntax

DELAY 45

### Description

Specifies the amount of time (in seconds) before a subtask attempts restarting itself following a termination. The maximum delay time allowed is 86,400 seconds (one day). Each subtask parameter card can be coded with its own unique delay time. Customize any subtask by entering in the desired DELAY card immediately before the subtask card. Any DELAY value entered becomes the default for all subsequent subtask cards.

The delay time reflects how quickly a needed resource can be recovered. A TCP subtask may require time to reset the port through which the workstation is connected. The default DELAY value is 30 seconds.

## FILTER Parameter Card

Identify the z/OS commands that may be issued to the mainframe agent.

### Valid Parameters

CMD *regexpression*

**Sample Syntax**

```
FILTER CMD ^D TCPIP, .*, NETSTAT, ROUTE$
FILTER CMD ^D NET, MAJNODES$
```

**Description**

Use this card to identify z/OS (MVS) commands that may be issued through the EView/390z extended console (initialized by the CMD parameter card above). Specify only one command expression per FILTER CMD line. If a command is issued which has no match in the command filter table, an EVO161 message will be returned to inform the caller that the command is unauthorized. Note that if no FILTER CMD cards are specified, then all commands will be forwarded to the console with no restrictions.

**Parameters**

<i>regexpression</i>	A Unix-style regular expression. An incoming command must match one of the regular expressions in the table of command filters before it will be sent to the console for execution. Be sure to use correct characters for the caret and square brackets if those characters are used in the <i>regexpression</i> . In the default IBM-1047 codeset, the values are:	
	<u>Character</u>	<u>EBCDIC Hexadecimal Value</u>
	Caret	^ x'54'
	Left Square Bracket	[ x'AD'
	Right Square Bracket	] x'BD'
	(Depending on the character mapping of the terminal emulator, the caret symbol may be displayed as a “not” ¬ symbol.)	

## NLS Parameter Card

Set the National Language Support (NLS) codeset value.

**Valid Parameters**

```
[CODESET=value] [LANG=langcode]
```

**Sample Syntax**

```
NLS CODESET=IBM-1047
NLS CODESET=IBM-939 LANG=JPN
```

**Description**

This card is used to identify the character set used on the mainframe and language used for supported message translations. The *value* must be a codeset provided by the z/OS Language Environment. A list of codeset values is provided in Appendix D of the *IBM C/C++ Programming Guide* (IBM publication SC09-4765). The *langcode* must be a supported language code that is available on this mainframe.

**Parameters**

<i>value</i>	The name of the codeset for the locale of the mainframe. The default is the EBCDIC "IBM-1047" codepage.
--------------	---

*langcode*

The 3-character language code used for message translations.  
The default is "ENG" (English).

## OSINFO Parameter Card

Initializes the OSI subtask to respond to various requests for z/OS Operating System information and statistics.

### Valid Values

SDSFMAX=*n*

### Sample Syntax

OSINFO SDSFMAX=400

### Description

Use the OSINFO card to initialize a subtask which will accept command type 46 requests from the EView/390z client and perform the requested function. See the *EView/390z Administrator's Reference* for syntax of type 46 requests and the available options.

### Parameters

*n*

An integer value indicating the maximum number of lines of information that will be returned from the queries to SDSF. Each line will contain information about one job. The default is 1000.

## PRINTCARDS Parameter Card

Write the SYSIN cards to the SYSPRINT.

### Valid Values

None

### Sample Syntax

PRINTCARDS

### Description

The PRINTCARDS card instructs the VP390 job to print each of the SYSIN lines that it reads to the SYSPRINT, excluding blank lines and comment lines. This is usually only used to document the cards that have been read when sending the SYSPRINT output to support for problem analysis. Enter PRINTCARDS as the first line of the SYSIN dataset member to document all the lines of the SYSIN input.

The PRINTCARDS card can be used multiple times in the SYSIN as a toggle control. The second occurrence of PRINTCARDS will stop the writing to SYSPRINT, the third occurrence will resume the writing, and so forth.

### Parameters

None

## MQMODEL Parameter Card

## MQCOMMAND Parameter Card

## MQDYNAMIC Parameter Card

Overrides to the default MQ Series queue names

### Valid Values

Valid MQ Series queue names (up to 48 characters)

### Sample Syntax

```
MQMODEL  EVIEW.COMMAND.REPLY.MODEL
MQCOMMAND EVIEW.COMMAND.INPUT
MQDYNAMIC EVIEW.COMMAND.REPLY.Q
```

### Description

These three parameter cards define overrides to the default MQ queue names when sending a command to the MQ Series. (See "Using OSINFO System Information API Commands" option 50 on page 24 of the *Administrator's Reference*.) The default queue names are:

Reply Model	SYSTEM.COMMAND.REPLY.MODEL
Command Queue	SYSTEM.COMMAND.INPUT
Dynamic Output Queue	EVIEW.COMMAND.REPLY.Q

If any of these default names are changed, the associated RACF permission must be changed to match the new names. (See "WebSphere MQ" on page 34.)

If used, these MQ parameter cards must be placed in the SYSIN deck ahead of the OSINFO card.

## RESTART Parameter Card

Number of restart attempts to allow a subtask before giving up.

### Valid Values

1 to 65535, or UNLIMITED

### Sample Syntax

```
RESTART 100
RESTART UNLIMITED
```

### Description

Specifies the number of times a subtask attempts to automatically restart. After this limit is reached, the subtask remains in a "down" state until it is manually reactivated using the `INIT` command. (See the description of the `INIT` command in Appendix A of the *EView/390z Administrator's Reference*.) Specify `UNLIMITED` instead of a number to allow a subtask to make an unlimited number of restart attempts. Each subtask can have a unique restart count by specifying another `RESTART` card immediately before the card which defines the subtask. The default `RESTART` value is 5.

## TCP Parameter Card

Identify port numbers and parameters for the TCP/IP connection to the EView/390z client.

### Valid Values

```
mmsport cmdport [hlq] [BUFDD=dd1,dd2 [ACK=ack] [LIMIT=limit]] [HB=hb]
[BINDIP=bindaddr] [SERVERIP=servaddr[/{maskaddr|maskprefix}]]
```

### Sample Syntax

```
TCP 6106 6107 BUFDD=BFR1,BFR2 ACK=5 LIMIT=20 HB=30 BINDIP=10.1.1.8 SERVERIP=10.1.1.0/24
```

### Description

This card will initialize a TCP subtask, which is responsible for opening two TCP/IP ports on the mainframe, then waiting for an EView/390z client component to start communication with the mainframe agent via these ports. While it waits for a connection, the TCP subtask can optionally write new mainframe messages to a set of buffering files, and then send the buffered messages after a connection is established. If you are also using the same EView/390z job for Discovery and Operations Management, you will need to define two TCP parameter cards. The *mmsport* and *cmdport* parameters must be unique for each TCP card defined.

### Parameters

<i>mmsport</i>	Port number opened on the mainframe for establishing a socket connection with the Master Message Server task on the EView/390z client. This number must match the EVOMF_HCI_AGENT_PORT value entered when adding the S/390 node through the EView/390z Task Manager.
<i>cmdport</i>	Port number opened on the mainframe for establishing a socket connection with the Command Server task on the EView/390z client. This number must match the EVOMF_CMDS_AGENT_PORT value entered when adding the S/390 node through the EView/390z Task Manager.
<i>hlq</i>	Optional, deprecated. High-level qualifier (hlq) for the mainframe TCP/IP datasets. This parameter is used to find the TCP/IP profile datasets, and is needed only if the default hlq is not used during TCP/IP installation. The <i>hlq</i> must be identified on each card. The TCP subtask will not be initialized if the <i>hlq</i> is misstated.
<i>dd1, dd2</i>	Optional. The DD names of the two buffering datasets. These DD names must be listed in the VP390 startup job, and they must point to predefined datasets with DCB=(DSORG=PS,RECFM=V,LRECL=1663).

<i>ack</i>	Optional. The number of unsolicited mainframe messages that will be passed on to the EView/390z client before an acknowledgment is expected from the EView/390z client. By default, EView/390z will expect an acknowledgment after every 5 messages. If an acknowledgment is not received, the mainframe agent will resend all of the messages back to the last successful acknowledgment. Then, if the EView/390z client acknowledgment is still not received, the mainframe agent will close the TCP/IP connection and wait for a reconnect request. Upon reconnection, all unacknowledged messages back to the last successful acknowledgment will be resent to the EView/390z client. The <code>ACK</code> parameter is only valid if <code>BUFDD</code> is specified
<i>limit</i>	Optional. The age limit (in minutes) of buffered messages that the mainframe agent will send to the EView/390z client. By default, messages read from the buffering files that are over 20 minutes old will not be forwarded to the EView/390z client. Set this value to 0 to receive all buffered messages regardless of their age. The <code>LIMIT</code> parameter is only valid if <code>BUFDD</code> is specified
<i>hb</i>	Optional. Length of time (in seconds) between heartbeat tests to verify the TCP/IP connection. By default, a short heartbeat message will be sent between the EView/390z client and the mainframe agent every 30 seconds.
<i>bindaddr</i>	Optional. An IPV4 dotted decimal address that names a specific mainframe IP address that the listening ports should bind to (useful at sites with multiple TCP/IP addresses defined in the same mainframe LPAR).
<i>servaddr</i>	Optional. An IPV4 dotted decimal address that names a specific UCMDB probe IP address that the listening ports will accept a connection from. <i>servaddr</i> must be specified in the nnn.nnn.nnn.nnn IP address format. Additionally, the <i>servaddr</i> can be masked to allow a range of IP addresses to connect to the ports. To specify a mask, add a slash (/) after <i>servaddr</i> followed by either a dotted decimal <i>maskaddr</i> (e.g., 255.255.255.0) or a numerical <i>maskprefix</i> between 0-32 to represent the number of bits to be included in the mask starting from the leftmost bit (e.g.: "16" would be equivalent to 255.255.0.0).

# Security and Discovery Requirements

This chapter describes the various security and other requirements for the different mainframe components to be discovered.





## CICS

The command consoles created by the EView Discovery agent must be defined to CICS unless auto-install has been configured for EMCS consoles in CICS.

Enter the following commands in CICS to define consoles that EView/390z can use to issue commands for information gathering:

```
CEDA DEF TE (CNxx) GR (EVOGRP) TY (DFHCONS) CONSN (staticname)  
CEDA DEF TE (CNyy) GR (EVOGRP) TY (DFHCONS) CONSN (EVRXCNzz)
```

where:

- EVOGRP** a group name for EView/390z.
- CNxx, CNyy** an available terminal name, for example CN10.
- staticname** the console name of the extended console defined on EView/390's CMD parameter card (see "CMD Parameter Card" definition page 24).
- zz** the two-character value of the &SYSCONE. system symbol for this LPAR. The resulting "EVRXCNzz" console will be dynamically created as needed in Rexx programs.

Run the install transactions after defining the consoles.

```
CEDA INSTALL GR (EVOGRP) TE (CNxx)  
CEDA INSTALL GR (EVOGRP) TE (CNyy)
```

See also the notes on defining EView/390z extended consoles to RACF in Appendix A.

If CICS Transaction and Program discovery is being performed, the EView discovery agent needs access to two modules in the CICS SDFHLOAD dataset (DFHCSDUP and DFHEITCU). There are three options to providing the agent access to these modules.

1. Add the SDFHLOAD dataset to the EView discovery agent STEPLIB DD statement. If this option is chosen then the SDFHLOAD dataset must be APF authorized.
2. Add SDFHLOAD to the LNKLIST.
3. Copy members DFHCSDUP and DFHEITCU from the SDFHLOAD dataset to the EV390.V63.LOAD dataset. This option should only be used if options 1 or 2 are not possible.

## DB2

Each DB2 subsystem that will be discovered will need to grant DISPLAY authority to the user or group that the EView/390z job is running under. See the DB2 section of Appendix A for the DB2 or RACF resource names to be modified.

# z/OS Console Authority

If OPERCMDS is active in RACF (or equivalent third-party security product), then the EView agent user must be given access to issue certain commands. The table in Appendix A shows the resource class permissions required for the console to perform discovery commands.

An additional level of command security for the agent command console is available through the use of the FILTER CMD parameter card. (See “FILTER Parameter Card” on page 25 for details on using command filtering.)

Some discovery scripts use REXX commands to perform discovery functions. If the REXX program issues z/OS commands, it will create a console with the name **EVRXCNxx**, where **xx** is the value in the system symbol &SYSCLONE. The table in Appendix A also includes resource class permissions required for the REXX command consoles.

## IMS

IMS DB/DC discovery requires that IMS MTO commands be entered through either the EMCS console interface or via the IMS outstanding reply message. If the IMS CMDMCS parameter is other than “N” then by default IMS commands issued by discovery scripts will be issued through the IMS Subsystem Interface (SSI) using the IMSID as the command prefix. If CMDMCS is specified as “N” then commands will be issued using the IMS outstanding reply message.

IMS DBCTL discovery requires that commands must be issued through the EMCS console using the IMS Subsystem Interface (SSI) using the IMSID as the command prefix. For discovery of IMS DBCTL, the CMDMCS parameter must be other than “N”.

## WebSphere MQ

If RACF security is active for WebSphere MQ, the following steps must be done depending on the RACF MQ classes that are active.

1. If the MQCONN class is active, the EView agent user must have access to the *mqssid*.*BATCH* profile (where *mqssid* is the MQ subsystem name):

```
PERMIT mqssid.BATCH CLASS (MQCONN) ID (agent-user) ACCESS (READ)
```

2. If the MQQUEUE class is active for WebSphere MQ, the agent user must have update access to the *SYSTEM.COMMAND.REPLY.MODEL* queue, for example:

```
PERMIT mqssid.SYSTEM.COMMAND.REPLY.MODEL CLASS (MQQUEUE) ID (agent-user) ACCESS (UPDATE)
```

and the command input queue:

```
PERMIT mqssid.SYSTEM.COMMAND.INPUT CLASS (MQQUEUE) ID (agent-user) ACCESS (UPDATE)
```

A profile for the EView/390z command reply queue must be created, for example:

```
RDEFINE mqssid.EVIEW.COMMAND.REPLY.Q UACC (NONE)
```

and the agent user given Alter access to create and delete this queue:

```
PERMIT mqssid.EVIEW.COMMAND.REPLY.Q CLASS(MQQUEUE) ID(agent-user)
ACCESS(ALTER)
```

3. If the MQCMDS class is active, the agent user must be given access to the following MQ display commands:

```
PERMIT mqssid.DISPLAY.SYSTEM CLASS(MQCMDS) ID(agent-user) ACCESS(READ)
PERMIT mqssid.DISPLAY.GROUP CLASS(MQCMDS) ID(agent-user) ACCESS(READ)
PERMIT mqssid.DISPLAY.QUEUE CLASS(MQCMDS) ID(agent-user) ACCESS(READ)
PERMIT mqssid.DISPLAY.QMGR CLASS(MQCMDS) ID(agent-user) ACCESS(READ)
PERMIT mqssid.DISPLAY.CHANNEL CLASS(MQCMDS) ID(agent-user) ACCESS(READ)
PERMIT mqssid.DISPLAY.CHINIT CLASS(MQCMDS) ID(agent-user) ACCESS(READ)
PERMIT mqssid.DISPLAY.GROUP CLASS(MQCMDS) ID(agent-user) ACCESS(READ)
```

The EView/390z discovery agent JCL must be updated to include the MQ SCSQAUTH dataset in the STEPLIB concatenation. While it is typical in some environments to create unique SCSQAUTH datasets to hold different startup parameter members (CSQZPARM), the only required dataset for the agent STEPLIB is the IBM-supplied SCSQAUTH dataset. SCSQAUTH datasets that contain only parameter members are not required in the STEPLIB.



# Starting and Stopping the Mainframe Component

This chapter explains how to start and stop EView/390z's VP390 job.

## Running EView/390z as a Started Task

The EView/390z job “VP390” may be run as a started task.



If you are monitoring multiple mainframe LPARs, the EView/390z job must be run with the same job name on all LPARs, or add an identifier name (which can access the jobs via the modify console command) to the job when starting that is the same on all LPARs.

### To Start the VP390 Job as a Started Task

To start VP390 as a task, follow these steps:

1. Copy the *hlq.SAMP (VP390)* procedure into the started tasks library.
2. Modify the dataset names according to the instructions at the top of the job.
3. Start the VP390 procedure from a z/OS console with the following command:

```
S VP390
```

### To Stop the VP390 Task

To stop the VP390 task, enter the following command from a z/OS console:

```
P VP390
```

## Running EView/390 as a Batch Job

The EView/390 job “VP390” may be run as a batch job.

### To Start the VP390 Job as a Batch Job

To start the VP390 as a batch job, modify and submit the JCL in

```
hlq.SAMP (VP390JCL) .
```

### To Stop the VP390 Batch Job

To stop the VP390 batch job, enter the following command from the operator console:

```
P VP390
```



## RACF Resource Class Permissions

This appendix identifies the resource class permissions required for the EView/390z consoles to perform their discovery commands.

# Security Settings for z/OS Resources

This table shows the commands that are issued by EView/390z to collect the Discovery information, and the security configuration changes needed for the user ID or group which EView/390z is running under.

Platform or Application	Commands Issued	RACF General Resource	Native or RACF Security Configuration Requirements
CICS	F <i>jobname</i> ,CEMT I SYSTEM	TCICSTRN GCICSTRN VCICSCMD	Class = TCICSTRN Profile = <b>SSID</b> .CEMT with Authority READ  Class = GCICSTRN Profile = CAT2 with Authority READ  Class = VCICSCMDS Profile = CEMT with Authority READ Member = INQUIRY
EView/390z Discovery Agent	F <i>jobname</i> ,SHOW VERSION	OPERCMD5	Class = OPERCMD5 Profile = MVS.MODIFY.STC.*.* with Authority READ Profile = MVS.MODIFY.JOB.* with Authority READ
DB2	<i>SubsystemPrefix</i> DISPLAY DDF	DSNR  MDSNSM	With native DB2 security:  Grant authority to DISPLAYAUTH group  Class = DSNR Profile = <b>DB2SSID</b> .BATCH with Authority READ  With RACF/DB2 security:  Grant authority to DISPLAYAUTH group  Class = DSNR Profile = <b>DB2SSID</b> .BATCH with Authority READ  Class = MDSNSM Profile = <b>DB2SSID</b> .DISPLAY with Authority READ
	<i>SubsystemPrefix</i> DISPLAY GROUP	DSNR  MDSNSM	With native DB2 security:  Grant authority to DISPLAYAUTH group  Class = DSNR Profile = <b>DB2SSID</b> .BATCH with Authority READ  With RACF / DB2 security:  Grant authority to DISPLAYAUTH group  Class = DSNR Profile = <b>DB2SSID</b> .BATCH with Authority READ  Class = MDSNSM Profile = <b>DB2SSID</b> .DISPLAY with Authority READ
	SELECT * FROM SYSIBM.LOCATIONS; SELECT * FROM SYSIBM.SYSDATABASE; SELECT * FROM SYSIBM.SYSTABLESPACE;	DSNR	With native DB2 security:  Grant authority to DISPLAYAUTH group  Class = DSNR Profile = <b>DB2SSID</b> .BATCH with Authority READ



		MDSNTB	<p>With RACF / DB2 security:</p> <p>Grant authority to DISPLAYAUTH group</p> <p>Class = DSNR Profile = <b>DB2SSID</b>.BATCH with Authority READ</p> <p>Class = MDSNTB Profile = <b>DB2SSID.OWNER.TABLE.ACTION</b> with Authority READ e.g.: Profile = <b>DB2SSID</b>.*.*.SELECT or Profile = <b>DB2SSID</b>.SYSIBM.*.SELECT or Profile = <b>DB2SSID</b>.SYSIBM.LOCATIONS.SELECT (access required must be granted to the most specific RACF rule defined)</p>
IMS (full)	<p>DISPLAY DATABASE ALL DISPLAY AREA ALL DISPLAY ACTIVE DISPLAY MODIFY ALL DISPLAY OLDS DISPLAY POOL ALL DISPLAY STATUS</p>	OPERCMD5	<p>With native IMS security:</p> <p>Exit = DFSCCMD0</p> <p>TABLE = ICMDTABL or create new entry VCMDTABL which only has the DIS command</p> <p>With RACF/IMS Security:</p> <p>Class = OPERCMD5 Profile = IMS.<b>IMSSSID</b>.DIS with Authority READ</p>
	RMLIST DBRC='LOG ALLODS'	OPERCMD5	<p>With native IMS security:</p> <p>Exit = DFSCCMD0</p> <p>TABLE = ICMDTABL or create new entry VCMDTABL which only has the DIS command</p> <p>With RACF/IMS security:</p> <p>Class = OPERCMD5 Profile = IMS.<b>IMSSSID</b>.RML with Authority READ</p>
IMS (DBCTL only)	<p>DISPLAY DATABASE ALL DISPLAY AREA ALL DISPLAY ACTIVE DISPLAY MODIFY ALL DISPLAY OLDS DISPLAY POOL ALL DISPLAY STATUS</p>	CIMS	<p>With native IMS security:</p> <p>Exit = DFSCCMD0</p> <p>TABLE = ICMDTABL or create new entry VCMDTABL which only has the DIS command</p> <p>With RACF/IMS Security:</p> <p>Class = CIMS Profile = DIS with Authority READ</p>
	RMLIST DBRC='LOG ALLODS'	CIMS	<p>With native IMS security:</p> <p>Exit = DFSCCMD0</p> <p>TABLE = ICMDTABL or create new entry VCMDTABL which only has the DIS command</p> <p>With RACF/IMS security:</p> <p>Class = CIMS Profile = RML with Authority READ</p>
MQ	<i>SubsystemPrefix</i> DISPLAY SYSTEM	MQCMDS	<p>Class = MQCMDS Profile = <b>MQSSID</b>.DISPLAY.GROUP with Authority READ</p>
	<i>SubsystemPrefix</i> DISPLAY QMGR ALL	MQCMDS	<p>Class = MQCMDS Profile = <b>MQSSID</b>.DISPLAY.QMGR with Authority</p>

			READ
	<i>SubsystemPrefix</i> DISPLAY CHINIT	MQCMDS	Class = MQCMDS Profile = <i>MQSSID</i> .DISPLAY.CHINIT with Authority READ
	<i>SubsystemPrefix</i> DISPLAY QUEUE (*), RNAME, RQMNAME, XMITQ, USAGE, DESCR, CLUSTER, CLUSNL, TARGQ, DEFTYPE, PROCESS	MQCMDS	Class = MQCMDS Profile = <i>MQSSID</i> .DISPLAY.QUEUE with Authority READ
	<i>SubsystemPrefix</i> DISPLAY CHANNEL (*), CHLTYPE, TRPTYPE, DESCR, CLUSTER, CLUSNL, CONNAME, XMITQ	MQCMDS	Class = MQCMDS Profile = <i>MQSSID</i> .DISPLAY.CHANNEL with Authority READ
MVS System Resources	D M=CPU	OPERCMD5	Class = OPERCMD5 Profile = MVS.DISPLAY.M with Authority READ
	D SYMBOLS	OPERCMD5	Class = OPERCMD5 Profile = MVS.DISPLAY.SYMBOLS with Authority READ
	D SSI	OPERCMD5	Class = OPERCMD5 Profile = MVS.DISPLAY.SSI with Authority READ
	D NET, MAJNODES	OPERCMD5	Class = OPERCMD5 Profile = MVS.DISPLAY.NET with Authority READ
	D ASM	OPERCMD5	Class = OPERCMD5 Profile = MVS.DISPLAY.ASM with Authority READ
	D PROD, STATE	OPERCMD5	Class = OPERCMD5 Profile = MVS.DISPLAY.PROD with Authority READ
	D PROD, REGISTERED	OPERCMD5	Class = OPERCMD5 Profile = MVS.DISPLAY.PROD with Authority READ
	D XCF, GRP	OPERCMD5	Class = OPERCMD5 Profile = MVS.DISPLAY.XCF with Authority READ
	D XCF, GRP, <i>GroupName</i> , ALL	OPERCMD5	Class = OPERCMD5 Profile = MVS.DISPLAY.XCF with Authority READ
	D TCPIP, <i>JobName</i> , NETSTAT, CONN D TCPIP, <i>JobName</i> , NETSTAT, ROUTE D TCPIP, <i>JobName</i> , NETSTAT, DEV D TCPIP, <i>JobName</i> , NETSTAT, ARP	OPERCMD5	Class = OPERCMD5 Profile = MVS.DISPLAY.TCPIP with Authority READ

## Extended Console Definitions in RACF

If RACF security (or equivalent security package) is active, the EView/390z agent console EVOCONSL must be defined to process commands on this extended console. Furthermore, if the EVORXCON Rexx function is expected to be used, the EVRXC�&SYSCLONE. name must also be defined to RACF.

Define the console names as users in RACF with an OMVS segment set to a non-zero UID, and define an OPERPARM segment with AUSTH=SYSTEM and CMDSYS=*LocalSystemName*. Use the RACF panels in TSO, a batch job, or the following commands to define the console names:

```
ADDUSER USER (EVOCONSL) DFLTGRP (xxxxxx) OPERPARM (AUTH (SYSTEM) ) MVS (UID (n) )
ADDUSER USER (EVRXCNyy) DFLTGRP (xxxxxx) OPERPARM (AUTH (SYSTEM) ) MVS (UID (n) )
```

where :

**EVOCONSL** the name of the static console defined on EView/390z CMD parameter card.

**xxxxxx** the group name to hold EView/390z consoles. This name can also be used in the GR() parameter when defining the consoles to CICS. A commonly used name is “EVOGRP”.

**yy** the &SYSCLONE. system symbolic, which is concatenated to “EVRXCN” to create a console name used by the EView/390z Rexx programs. Use the system console command “D SYMBOLS” to find the value of the &SYSCLONE. symbol.

**n** a non-zero OMVS UID.

If CICS discovery is to be attempted, the EVOCONSL and EVRXCNyy names must also be defined as terminals in CICS (see page 33).