EView/400i IBM i (iSeries-AS/400) Discovery for Micro Focus Universal Discovery (UD) for UCMDB

Administrator's Reference

Software Version: 6.3

January 2018

Legal Notices

Warranty

EView Technology, Inc. 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

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.
Raleigh, North Carolina 27616
United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Copyright Notices

Trademark Notices

EView/400i® is a registered U.S. trademark of EView Technology, Inc.

AS/400, OS/400, IBM i, and iSeries 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.

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

Printing History

The printing date and part number of the manual indicate the edition of the manual. The printing date will change when a new edition is printed. Minor changes may be made at reprint without changing the printing date.

Manual updates may be issued between editions to correct errors or document product changes. Contact EView Technology support to verify that you have the updated or new editions.

Table 1 indicates changes made to this document since the last released edition.

Table 1: Changes to This Document

Date	Description
January 2018	Version 6.3

Support

Visit the EView Technology web site at:

http://www.eview-tech.com/

This web site provides contact information and details about the products, services, and support that EView Technology offers.

You can also contact EView Technology support via e-mail. See the web site for contact information.

Contents

Table of Contents

1

Conventions

The following typographical conventions are used in this manual.

Table 1-1: Typographical Conventions

Font	Meaning	Example
Italic	Book or manual titles, and man page names	See the *EView/400i Administrator's Reference* for more information.
	Provides emphasis	You *must* follow these steps.
	Specifies a variable that you mus supply when entering a comman	At the prompt, enter rlogin *your_name* where you supply your login name.
	Parameters to a function	The *oper_name* parameter returns an integer response.
Bold	New terms	The **monitor agent** observes...
`Computer`	Text and items on the computer screen	The system replies: Press `Enter`
	Command names	Use the `grep` command ...
	Function names	Use the `opc_connect()` function to connect...
	File and directory names	`/opt/OV/bin/OpC/`
	Process names	Check to see if `opcmona` is running.
	Window/dialog box names	In the `Add Logfile` window...
`Computer Bold`	Text that you must enter	At the prompt, enter **`ls -l`**
`Keycap`	Keyboard keys	Press **`Return`**.
[Button]	Buttons on the user interface.	Click the [Apply] button.
Menu Items	A menu name followed by a colon :) means that you select the men then the item. When the item is followed by an arrow (->), a cascading menu follows.	Select Actions:Utilities->Reports ...

2

Documentation

EView/400i Discovery for IBM i (iSeries-AS/400) 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/400i documentation, related Micro Focus UCMDB products provide a comprehensive set of manuals that help you use and understai the products' underlying concepts.

EView/400i Printed Manuals

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

EView/400i Discovery for IBM i (iSeries-AS/400) Installation Guide

Explains how to install, de-install, and configure EView/400i. Also includes how to transfer installation files from the UCMBD server to the AS/400 agent.

EView/400i Discovery for IBM i (iSeries-AS/400) Administrator's Reference

Explains how to customize and use EView/400i. Also includes detailed troubleshooting procedures and explanations of EView/400i system messages.

EView/400i Online Information

The following information is available online:

- *EView/400i Discovery for IBM i (iSeries-AS/400) Installation Guide*
- *EView/400i Discovery for IBM i (iSeries-AS/400) Administrator's Reference*
- *EView/400i Discovery for IBM i (iSeries-AS/400) Software Release Notes*

3

Configuring EView/400i

This chapter describes how to configure EView/400i from the UCMDB Discovery probe and distribute the configurations to the AS/400 agents. Phase 1 must be done once on the UD management server.

Phases 2 – 3 should be done for each AS/400 node that is to be monitored.

Phase 1: Install Perl

EView/400i and the Micro Focus Universal Discovery (UD) adapter requires installation of ActivePerl on the OMW server (available as a free download at `http://www.activestate.com`). Download and install Perl before activating any iSeries discovery jobs.

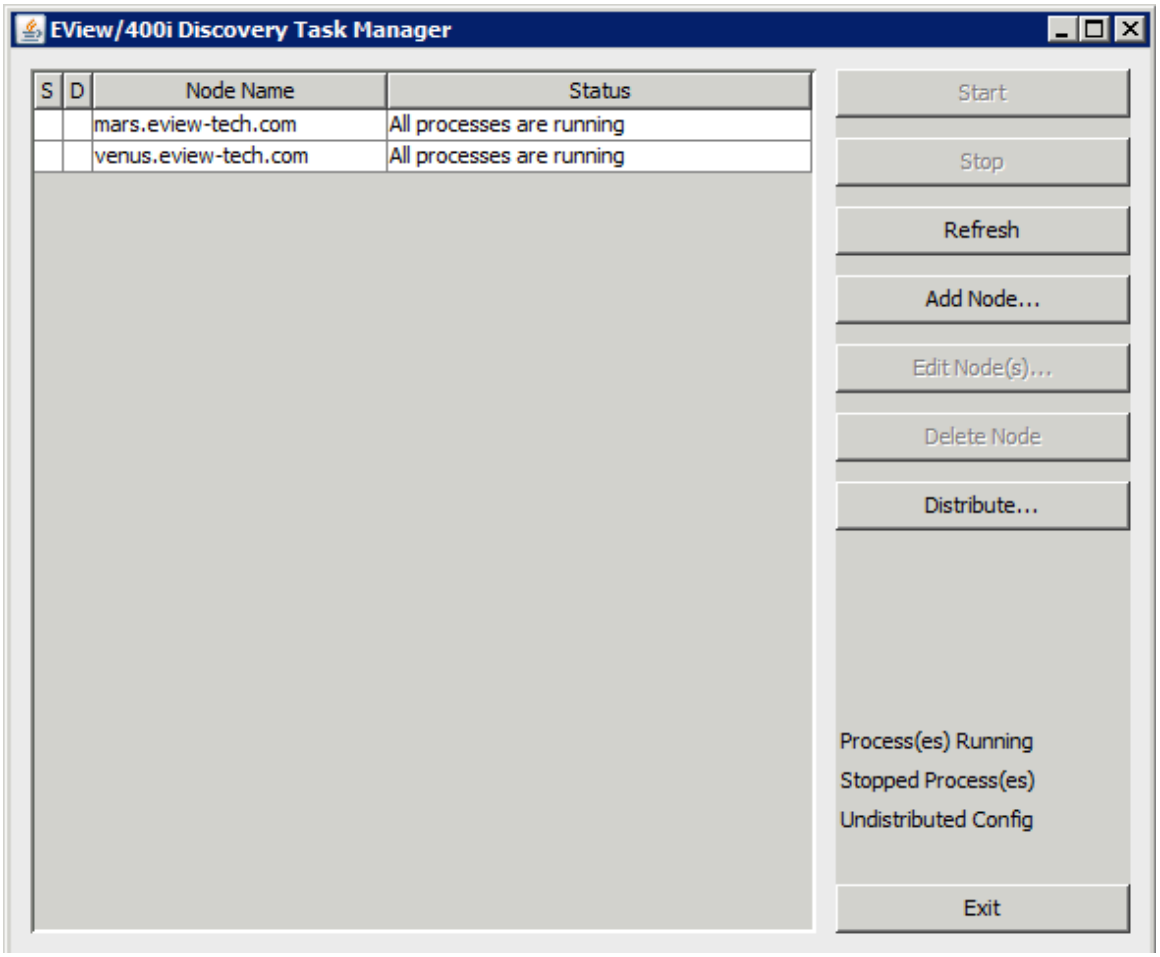
Phase 2: Add, Modify, and Distribute Agent Parameters

In this section, iSeries (AS/400) nodes are defined to the EView/400i client component. Agent parameters are distributed to the iSeries system.

Starting the iSeries Agent Configuration Interface

The EView/400i Task Manager is used for adding iSeries systems for discovery and configuring agent parameters.

Figure 3-10: EView/400i Task Manager



The EView/400i Task Manager can be started from the Windows Start->Programs->EView Technology->EView 400 menu.

Add iSeries Systems and Runtime Parameters

The EView/400i Task Manager will define iSeries systems to be discovered.

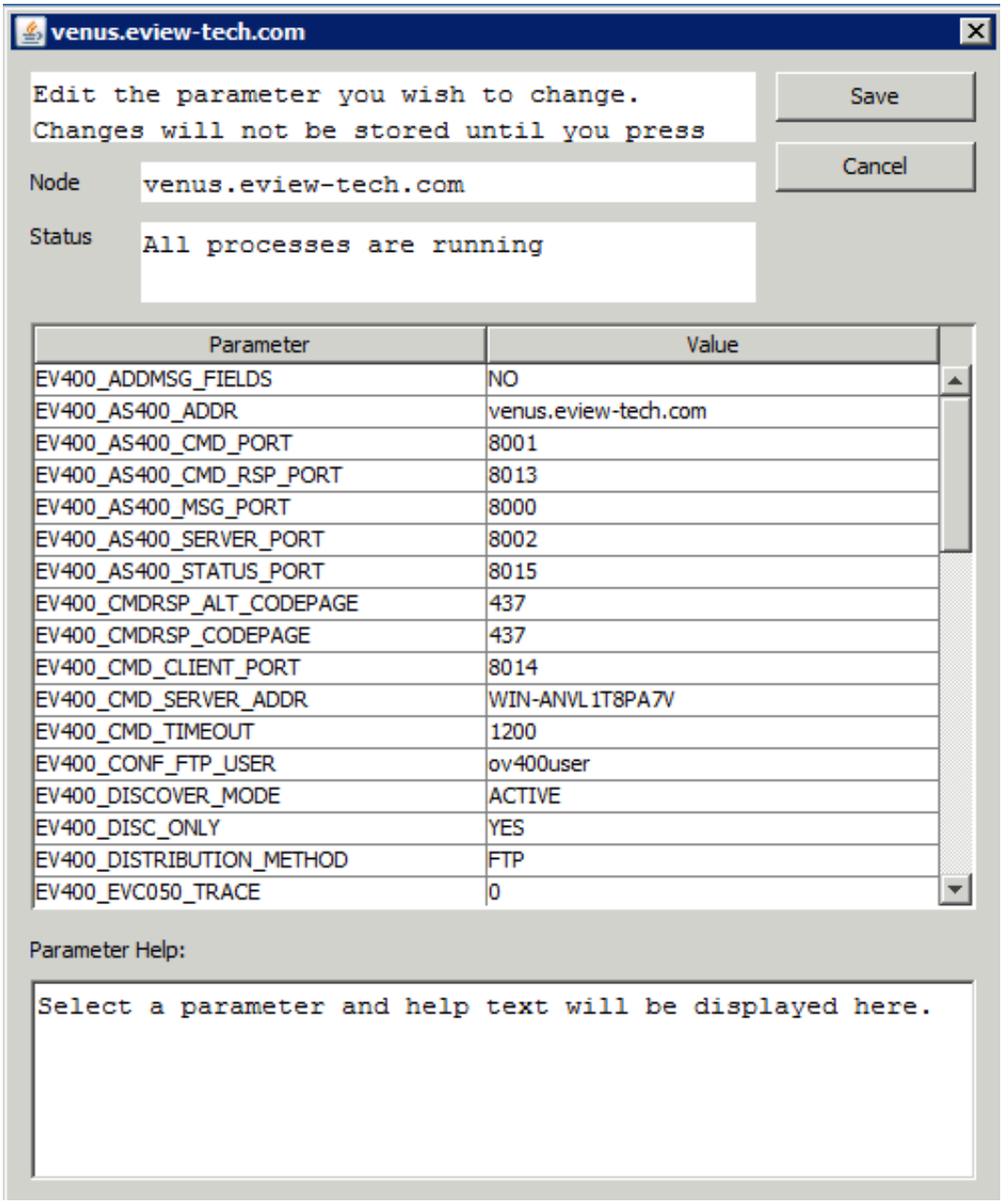
1. Start the EView/400i Task Manager.

   Start->Programs->EView Technology->EView 400->Task Manager

2. Select [Add Node] and enter the fully qualified name of the iSeries system to be discovered. The name must be able to be resolved through the existing name service (for example, DNS or an entry in the Windows' `system32\drivers\etc\hosts` file).

3. Select the new node name in the list and click [Edit Node] to modify any of the configuration parameters as explained below (or hovering the mouse pointer over the parameter name in the window).

Figure 3-13: EView/400i Node Configuration Parameters



The node configuration parameters are described in Table 3-1. Many of the parameters listed are not used when the agent is being used only for DDMA discovery. All parameters are listed as they are required for compatibility when the agent is used for Operations Management. **In environments where the agent is used for both Operations Management and discovery, the agent parameters should ONLY be distributed from the Operations Management server.**

Table 3-1: EView/400i Node Parameters

Parameter	Description	Valid Values	Default Value
EV400_ADDMSG_FIELDS (IS NOT USED FOR DISCOVERY)	Indicates whether EV/400 will se the Program Name and Message Type fields in the messages that are sent to the OMW.	YES – The EV/400 message serve will send the "Program Name" ar "Message Type" fields in its messages to the OMV These fields were added in the EV/400 Version 2.0 and will need to be accommodated in any existing template conditions that were written for Version 1.0 of the product. NO – Use this option if you are using message template condition from EV/400 Version 1.0 and do r wish to modify those existing templates to utilize the new fields.	NO
EV400_AS400_ADDR	Fully qualified network name of AS/400 where the EV/400 agent component is installed.	Name of AS/400 managed node.	None
EV400_AS400_CMD_PORT	TCP port number assigned to the EV/400 Command Server process.	Any unused port number on the AS/4 agent between 1025 and 65535.	8001
EV400_AS400_CMD_RSP_PORT	TCP port number assigned for communication between the EV/400 Message Server Process and Command Server Process	Any unused port number on the OM server between 1025 and 65535.	8003*
EV400_AS400_MSG_PORT	TCP port number assigned to the EV/400 Master Message Server process.	Any unused port number on the AS/4 agent between 1025 and 65535.	8000
EV400_AS400_SERVER_PORT	A TCP port number reserved for inte process communications on the AS/400 agent.	Any unused port number on the AS/4 agent between 1025 and 65535.	8002
EV400_CMD_CLIENT_PORT	A TCP port number used by the Command Server process to communicate with the Master Messa Server process. This port number mu be unique on the OMW server where the Command Server and Master Message Server processes are running.	Any unused port number on the OM server between 1025 and 65535.	8004*
EV400_CMD_SERVER_ADDR	The name of the OMW server where the Command Server process is to run.	A Windows server name.	The OMW server nar
EV400_CMD_TIMEOUT	The amount of time to wait for an AS/400 command response (in seconds).	An integer greater than or equal to 1 (second).	30
EV400_CONF_FTP_USER	A user ID on the AS/400 agent that has the authority to store configuration files in the EVIEW library. (See note on page 17.) If this field is left blank, the EV/400	An AS/400 user ID (up to 10 characters) or blank.	ov400user

	configuration utility will prompt you for a user ID when it is making its ftp connection to the AS/400.		
EV400_DISTRIBUTION_METHOD	The method to use for distributing configuration data to the AS/400 nod	FTP – Use ftp for distribution. HOST – Use AS/400 Host Services. The AS/400 Host Services job must be running on the AS/400 agent.	FTP
EV400_LICKEY	License key for the managed node. To obtain a license key, contact EView Technology support at support@eview-tech.com. (See page 13 of the *EView/400i Installati Guide*.)	Valid license key	None
EV400_MONITOR_QHST **(IS NOT USED FOR DISCOVERY)**	Indicates whether the EV/400 agent should monitor for messages tl are sent to the AS/400 QHST system history log. If set to "YES", then veri that the EV400_QHST_MON_FREQ field is greater than 0.	YES or NO	YES
EV400_MON_AUDJRNL **(IS NOT USED FOR DISCOVERY)**	A list of two-character entry types fr the QAUDJRN that should be forwarded from the AS/400 agent. Entry types are separated by comma See AS/400 documentation (such as t *iSeries Security Reference*) or Appenc E for descriptions of journal entry type	AD,AF,AU,CA,CD,CO,CP,DO,DS, NA,OW,PA,PG,PW,ST,SV,VA,VP, VU,ZC,ZR ALL – All of the above NONE – None of the above	NONE
EV400_MON_RESOURCES **(IS NOT USED FOR DISCOVERY)**	Indicates whether the EV/400 agent should monitor the status of AS/400 resources (lines, controller and devices). This function is not use in EView/400i for Windows and should always be s to "NO".	YES or NO	NO
EV400_MSG_DISTRIB **(IS NOT USED FOR DISCOVERY)**	Should the AS/400 agent send its collected messages to all OMW serve that are in contact with it? (If "NO", then specify in the EV400_PRIMARY_SERVER field which OMW server is the primary recipient of messages.)	YES – Send unsolicited AS/400 messages to all EV/400 servers that are in contact with this agent. NO – Send unsolicited messages only to the primary server.	YES
EV400_MSG_SERVER_ADDR	The name of the Windows server wh the Master Message Server process i to run.	A Windows server name	The OMW server
EV400_PATH **(IS NOT USED FOR DISCOVERY)**	The EV/400 installation directory on the OMW server.	EV/400 home directory	\Program Files\EVi Technolog EView 40(
EV400_PERF1 **(IS NOT USED FOR DISCOVERY)**	Specifies whether the performance gathering function will be activated on the AS/400 agent to gather the data for performance grou 1. See Appendix D for the list of metrics collected in group 1.	YES – Activate the performance gathering function on the AS/400 agent. NO – Do not activate performance da gathering for group 1.	NO

EV400_PERF1_INT **(IS NOT USED FOR DISCOVERY)**	The interval, in minutes, at which group 1 performance data is collected on the AS/400 agent and forwarded to the OMW server. This field is only needed if EV400_PERF1 is set to "YES".	An integer greater than or equal to 1 (minute).	5
EV400_PERF2 **(IS NOT USED FOR DISCOVERY)**	Specifies whether the performance gathering function will be activated on the AS/400 agent to gather the data for performance grou 2. See Appendix D for the list of metrics collected in group 2.	YES – Activate the performance gathering function on the AS/400 agent. NO – Do not activate performance da gathering for group 2.	NO
EV400_PERF2_INT **(IS NOT USED FOR DISCOVERY)**	The interval, in minutes, at which group 2 performance data is collected on the AS/400 agent and forwarded to the OMW server. This field is only needed if EV400_PERF2 is set to "YES".	An integer greater than or equal to 1 (minutes).	30
EV400_PRIMARY_SERVER **(IS NOT USED FOR DISCOVERY)**	The fully qualified name of the prim OMW server to receive messages fro this agent. Although multiple OMW servers may be connected to the AS/400 ager at one time, only the server named here will receive unsolicited AS/400 messages. This field is only necessar when the EV400_MSG_DISTRIB parameter is "NO".	An EV/400 server name	null
EV400_QHST_MON_FREQ **(IS NOT USED FOR DISCOVERY)**	Frequency (in seconds) that the EV/ agent collects new messages from the QHST system history log. This field is only necessa when the EV400_MONITOR_QHST parameter is "YES".	An integer greater than or equal to 1 (seconds)	5
EV400_USE_MWA **(IS NOT USED FOR DISCOVERY)**	Indicates whether OVPA is installed and being used for storing collected AS/400 performance data.	YES – OVPA is installed and being used. NO - OVPA is not installed.	NO
EV400_WORK_AREA **(IS NOT USED FOR DISCOVERY)**	Specifies where EV/400 places temporary work files on the OMW server.	Any existing directory on the OMW server	\temp
EV400_VP400CS_TRACE	Set tracing level for the command server (ev400cs). Multiple values can be added together in hexadecima	0 - No tracing output enabled 0001 - general program trace enabled 0002 - internal tracing enabled 0004 - program detail tracing enabled 0008 - warning messages enabled 0010 - error tracing enabled 0020 - dump output enabled 0040 - loop tracing enabled 0080 - verify tracing enabled	0.
EV400_VP400MMS_TRACE	Set tracing level for the master message server (ev400mms) . Multip values can be added together in hexadecimal.	0 - No tracing output enabled 0001 - general program trace enabled 0002 - internal tracing enabled 0004 - program detail tracing enabled 0008 - warning messages enabled 0010 - error tracing enabled	0

		0020 - dump output enabled 0040 - loop tracing enabled 0080 - verify tracing enabled 0100 - log messages sent to OMW 0200 - log performance records	
EV400_VP400HOSTCMD_TRACE	Set tracing level for the host commar client `ev400hostcmd` process. Multiple values can be added togethe in hexadecimal.	0 - No tracing output enabled 0001 - general program trace enabled 0002 - internal tracing enabled 0004 - program detail tracing enabled 0008 - warning messages enabled 0010 - error tracing enabled 0020 - dump output enabled 0040 - loop tracing enabled 0080 - verify tracing enabled	0
EV400_EVCMSG_TRACE	Set tracing level for the agent messa TCP task (EVCMSG). Multiple valu can be added together in hexadecima	0 - No tracing output enabled 0001 - general program trace enabled 0002 - internal tracing enabled 0004 - program detail tracing enabled 0008 - warning messages enabled 0010 - error tracing enabled 0020 - dump output enabled 0040 - loop tracing enabled 0080 - verify tracing enabled	0
EV400_EVCHCI_TRACE	Set tracing level for the agent messa transfer process (EVCHCI)	0 - disables tracing 1 - enables tracing	0
EV400_EVC050_TRACE	Set tracing level for the agent command processor (EVC050)	0 - disables tracing 1 - enables tracing	0
EV400_EVCQSCAN_TRACE	Set tracing level for the agent messa queue monitor (EVCQSCAN)	0 - disables tracing 1 - enables tracing	0
EV400_EVPERFM_TRACE	Set tracing level for the agent performance monitor process (EVPERFM)	0 - disables tracing 1 - enables tracing	0
EV400_EVCCTL_TRACE	Set tracing level for the API interface process (EVCCTL)	0 - disables tracing 1 - enables tracing	0
EV400_EVC070_TRACE	Set tracing level for the agent resour monitor (EVC070)	0 - disables tracing 1 - enables tracing	0
EV400_EVCCMD_TRACE	Set tracing level for the agent command TCP process (EVCCMD)	0 - disables tracing 1 - enables tracing	0
EV400_EVHSTPGM_TRACE	Set tracing level for the agent history log (QHST) monitor	0 - disables tracing 1 - enables tracing	0
EV400_VP400MMS_LOGSIZE	Set the maximum log size in 1K increments for the master message server (ev400mms)	1-99999 (kilobytes)	3000
EV400_VP400CS_LOGSIZE	Set the maximum log size in 1K increments for the command server (ev400cs)	1-99999 (kilobytes)	3000
EV400_CMDRSP_CODEPAGE	Set the code page to be used for converting command responses.	Any codepage supported by OM and Windows such as 1252 – Latin I 932 – Japanese Shift-JIS 936 – Simplified Chinese 949 – Korean A value of "UTF-8" indicates that command output is not converted usi any codepage. UTF-8 is the expected character set for the OM Windows to output window.	UTF-8

EV400_CMDRSP_ALT_CODEPAGE	Set an alternate code page to be used for converting command responses when EV400_CMDRSP_CODEPAGE is set to UTF-8. This parameter is only used with the ev400hostcmd option 81. If EV400_CMDRSP_CODEPAGE is not set to UTF-8, this parameter is ignored.	Any valid windows code page, but in most cases the default value of 437 would be used.	437
EV400_OVOMSG_CODEPAGE **(IS NOT USED FOR DISCOVERY)**	Set the code page to be used for converting messages that are sent to the OM message interface.	Any codepage supported by OM and Windows. 1252 – Latin I 932 – Japanese Shift-JIS 936 – Simplified Chinese 949 – Korean	1252
EV400_NLS_CCSID	Set the CCSID for the language libra that is being used as the subsystem library for the EView/400i agent subsystem.	Any CCSID supported on OS/400. Some typical values are: QSYS2924 English – 37 QSYS2928 French – 297 QSYS2929 German – 273 QSYS2931 Spanish – 284 QSYS2932 Italian – 280 QSYS2962 Japanese – 5026 QSYS2986 Korean – 933 QSYS2989 Simplified Chinese – 935	37

* This port number will be incremented automatically for new nodes that are added so that the port numbers remain unique on the server.

4. Save the parameters for this agent. The Node Configuration program will save the parameters locally on the Probe server.

5. (Multiple node definitions may be edited at once to edit the same parameters that are not required to be unique. Select the node names while holding down the **Shift** key then click the [Edit Node(s)] button.)

6. Select any nodes in the list of defined nodes that have the "Distributed?" box marked with a red slashed circle and click the [Distribute...] button to send the configuration parameters to the AS/400 agents. Multiple nodes may be selected at one time by holding down the **Shift** key while selecting the node names. If the EV400_CONF_FTP_USER field for a node has not been filled in, you will be prompted to enter the user ID.

The AS/400 User ID that is used in the distribution process must have the authority to create and modify the configuration objects in the EVIEW library, and have change authority over the nine configuration objects in the library: EVCMDAUD, EVF020, EVF021, EVJOBFILT, EVPARMS, EVMSGQCFG, EVFILTRSPC, EVQCFGSPC, and EVHSTFLSPC.

7. Select the line for the added node and click the [Start] button to start the EView/400i server processes for the AS/400 node. (The processes on the agent side will be started in Phase 3.)

Whenever a node's configuration parameters are changed using the [Add Node] or [Edit Node] functions of the Node Configuration program, the Distributed column will be marked to remind you that there are changes that need to be distributed to the AS/400 agent.

(If a node is deleted using the [Delete Node] button, the node will be removed from the EView/400i list of defined nodes, but it will need to be manually removed from the OMW list of managed nodes.)

Phase 3: Starting the Agent Processes

After the iSeries (AS/400) agent configuration parameters have been customized and distributed to the agents, start the EVSBS subsystem from the AS/400 agent with the command:

```
CALL EVIEW/EVINIT
```

Use the `WRKACTJOB` command to verify that the EVSBS subsystem is active.

Optional PARM values are available for the EVINIT command:

ALL	**Start all jobs in the subsystem that have been configured. This is the default option.**
CLEARQ	Clear any buffered messages from the EView message queues before starting the subsystem jobs.
TEST	Instead of starting the jobs, EVINIT will display the SBMJOB commands that would be used to start the jobs. This may be useful to verify that the jobs are being started with the desired options.
VERSION	Display the version of the installed agent software.
jobname	Start specific job(s) in the subsystem. If a job has fallen into a Message Wait status, use ENDJOB OPTION(*IMMED) to stop the individual job, then restart it by specifying the specific process name in the PARM when calling EVINIT. Job names are listed in Appendix C.

Examples:

To clear the agent's internal data queues before starting the EVSBS agent subsystem:

```
CALL EVIEW/EVINIT PARM(CLEARQ)
```

To start only the EVSHSTPROC, EVSTCPPROC, and EVTCTLPROC jobs:

```
CALL EVIEW/EVINIT PARM('EVSHSTPROC EVSTCPPROC EVTCTLPROC')
```

Stopping the EVSBS Subsystem

To terminate a running EView/400i subsystem on the AS/400 agent, use the command:

```
ENDSBS EVSBS *IMMED
```

The EVSBS subsystem must be ended prior to executing any save commands that would allocate an EView/400i object, such as when performing a backup.

4

Using EView/400i

This chapter describes how to use EView/400i Discovery client interface.

Sending Commands to the Agent

Commands may consist of native OS/400 commands as well as scripts or programs. To send a native OS/400 command, use the `ev400hostcmd` program located in the `\bin` subdirectory of the EView/400i installation path. The format of `ev400hostcmd` is:

`ev400hostcmd` ***`type command.as400_node`***

where:

type	The command type. Use one of the following types: **80** = Any response text (for example, from a Display command) is displayed using the OM server's default codepage. **81** = Any response data should be converted using the code page specified in the EV400_CMDRSP_ALT_CODEPAGE parameter (see parameter definition on page 17). **86** = a special EV/400 command that requests AS/400 information on certain system resources. See "Using the Agent Interface to System APIs" on page 24.
command	The OS/400 command text. The first period (.) encountered is used to mark the end of the command. If the command text itself has a period in it, enter two periods to signify that it is not the end of the command.
as400_node	The AS/400 node on which the command is to be executed. Usually this is the node which generated the original message: <$MSG_NODE_NAME>

Using the EVDSPPFM Command

The EVDSPPFM command in the EVIEW library can be called to display the contents of a physical file. This command differs from the operating system command DSPPFM in that it sends the output to *PRINT allowing the display of physical files using the `ev400hostcmd` interface from the OMW server.

The command syntax is:

`EVIEW/EVDSPPFM FILE(`*`library/filename`*`) MBR(`*`membername`*`)`

where:

filename	Name of the physical file
library	Library where ***filename*** resides
membername	Member name in physical file to list. For files with only one member, the ***membername*** is the same as the ***filename***.

Using the EView/400i Agent Interface to System APIs

The EVCCTLPROC job of the EV/400i agent provides a direct interface to retrieving operating system information through the use of system APIs without the need to execute OS/400 commands. The output of information retrieved in this manner is presented in a format that can be parsed by a script on the UD management server. Access to this API interface is requested using type 86 of the `ev400hostcmd` utility program. (See page 23 for the general syntax of the `ev400hostcmd`.) The type of data requested is specified by a two-digit code followed by a vertical bar (|) and additional parameter information depending on the selected code. The syntax of the ev400hostcmd with type 86 is:

ev400hostcmd 86 *code*[|*parameters*].*as400hostname*

where:

code	The two-digit code for the information options requested below
parameters	The additional parameters for the selected ***code***
as400hostname	The name of the AS/400 agent that is to receive the command

Keep in mind that if this command is entered from a Windows command line or script, vertical bars will need to be escaped by enclosing everything after the "86" in double quotation marks.

Sample Commands

1. To issue a code 09 request for jobs in the QPRINT output queue in library QGPL of the AS/400 named bluebox1.mycom.com, enter the command:

   ev400hostcmd 86 "09|QPRINT|QGPL.bluebox1.mycom.com"

   Output lines will be returned with values separated by a vertical bar. One line will be generated for each record found, in this case, a line representing each job. The last line will be the text "EOF".

2. To issue a code 42 request for "netstat" network connections of type UDP on the AS/400 named bluebox2.mycom.com, enter the command:

   ev400hostcmd 86 "42|||UDP.bluebox2.mycom.com"

The available codes are:

01 - Active Job Listing

Retrieves a list of active jobs.

Parameters: None

Output: One line for each active job found, in the following format:

Job name
User name
Job number
Internal job ID in printable hexadecimal
Job status (e.g., MSGW, DEQW, TIMW, EVTW, etc.)
Job type (A=autostart, B=batch, I=interactive, M=subsystem monitor, R=spooled reader, S=system, W=spooled writer, X=start-control-program-function system job)
Subsystem name
Run priority
Function type
Function name

Sample Output:

```
QTLPD03673|QTCP      |009959|00140001001273008 9B003C60A7AD6BE|TIMW|B|QSYSWRK   |25| |
QYPSJSVR  |QYPSJSVR |009965|0014000100128B0089B00444FCA4007A|SIGW|B|QSYSWRK   |10|P|QYPSJSVR
QSQSRVR   |QUSER    |009966|0014000100128F0089B004877099597E|CNDW|B|QSYSWRK   |10| |
QPADEV0006|QSECOFR  |010292|00140001000083008 9D32798E7C4A1D2|DSPW|I|QBASE     |20|C|WRKUSRPRF
QPADEV0005|USER1    |010123|0014000100008F0089BB52A5790BD50C|DSPW|I|QBASE     |20|C|DLTUSRSPC
QTFTP00830|QTCP     |010124|001400010000D70089BB580447A79194|DEQW|B|QSYSWRK   |25| |
EVSBS     |QSYS     |010258|001400010000E70089D0E0B30009C97E|DEQW|M|EVSBS     |0| |
```

02 - Job Attributes

Retrieves additional attributes of a requested job.

Parameters: Internal job ID in printable hexadecimal (field #4 of the code 01 output above)

Output: One line with the following format:

System pool ID
CPU used (milliseconds)
Auxiliary IO requests
Interactive transactions response time
Thread count
Date entered system (CYYMMDDHHMMSS), where:
- C – century, 0 indicates years 19xx and 1 indicates years 20xx.
- YY – Year
- MM- Month
- DD – Day
- HH – Hour
- MM – Minutes
- SS – Seconds

Date job active (CYYMMDDHHMMSS)
Job description name
Job description library
Submitter job name
Submitter user name

Sample Output:

```
3|1014|1079|5|7323|1|1050623110833|1050623110833|QDFTJOBD  |QGPL      | |
```

04 - List Subsystems

Returns a list of all active subsystems.

Parameters: None

Output: One line for each job queue, in the following format:

Subsystem name
Library in which subsystem's description resides
Maximum number of active jobs allowed in subsystem, or -1 if no maximum
Number of jobs currently active in the subsystem
Subsystem description text

Sample Output:

```
EVSBS     |EVIEW     |-1|10|
QBASE     |QSYS      |-1|11|BASIC CONTROLLING SYSTEM
QHTTPSVR  |QHTTPSVR  |-1|8|HTTP SERVER SUBSYSTEM
QSERVER   |QSYS      |-1|13|FILE SERVER SUBSYSTEM
QSPL      |QSYS      |-1|1|SPOOLING SUBSYSTEM
QSYSWRK   |QSYS      |-1|81|SYSTEM SUBSYSTEM
QUSRWRK   |QSYS      |-1|13|USER SUBSYSTEM
EOF
```

07 - Job Queue Request

Returns the list of job queues.

Parameters: None

Output: One line for each job queue, in the following format:

Job queue name
Job queue library
Number of jobs in job queue
Subsystem Name
Job Queue Status

Sample Output:

```
QBASE     |QGPL      |0|QBASE     |RELEASED
QBATCH    |QGPL      |0|QBASE     |RELEASED
QFNC      |QGPL      |0|          |RELEASED
QINTER    |QGPL      |0|QBASE     |RELEASED
QPGMR     |QGPL      |0|          |RELEASED
QSNADS    |QGPL      |0|          |RELEASED
QSPL      |QGPL      |0|QSPL      |RELEASED
QS36EVOKE |QGPL      |0|QBASE     |RELEASED
QS36MRT   |QGPL      |0|QBASE     |RELEASED
QTXTSRCH  |QGPL      |0|          |RELEASED
```

08 - Output Queue Request

Returns the list of output queues.

Parameters: None

Output: One line for each output queue, in the following format:

Output queue name
Output queue library
Number of files in queue
Writer
Output Queue Status

Sample Output:

```
QDKT       |QGPL       |0|          |RELEASED
QPFROUTQ   |QGPL       |0|          |RELEASED
QPRINT     |QGPL       |64|          |RELEASED
QSPRCLOUTQ |QRCL       |0|          |RELEASED
QEZDEBUG   |QUSRSYS    |0|          |RELEASED
QEZJOBLOG  |QUSRSYS    |18|          |RELEASED
QTPPPOUTQ  |QUSRSYS    |0|          |RELEASED
```

09 - Output Queue Listing

Returns a list of jobs in the specified output queue.

Parameters: (separated by vertical bar)

Output queue name
Output queue library

Output: One line for each spool file in the queue, in the following format:

Job name
User name (owner of the spooled file)
Job number
Spooled file name
Spooled file number
Spooled file status
Spooled file open date and time (CYYMMDDHHMMSS)
User-specified data
Form type
Total pages
Number of copies
Priority

Sample Output:

```
QPADEV0005|USER1     |010067|TSTRPI    |8|*READY    |1050601111933|CRTBNDC   |*STD      |6|1| 5
MAKEPGM   |USER1     |010071|MAKEPGM   |1|*READY    |1050601114425|          |*STD      |3|1| 5
EVACMDPROC|EVUSER    |009996|QPRINT    |1|*READY    |1050531150433|          |*STD      |1|1| 5
QPADEV0004|USER1     |010088|QPDZDTALOG|1|*READY    |1050602171437|          |*STD      |1|1| 5
QPADEV0005|USER1     |010123|QPDZDTALOG|3|*READY    |1050604124551|          |*STD      |1|1| 5
QPADEV0003|USER2     |010154|QPUOPRTF  |2|*READY    |1050608165059|          |*STD      |2|1| 5
QPADEV0003|USER2     |010246|QSYSPRT   |1|*READY    |1050621110111|          |*STD      |1|1| 5
```

10 - List Objects

Returns a list of objects from one or more libraries

Parameters: (separated by vertical bars)

Object Name (or *ALL)
Library Name (or *ALL)
Object Type (or *ALL)

NOTE: *ALL cannot be used for all three parameters.

Output: One line for each object found on the system in the format:

Object Name
Library Name
Object Type
Creation Date and Time (MM/DD/YYYY HH:MM:SS)
Last Changed Date and Time (MM/DD/YYYY HH:MM:SS)
Creator's User Profile
Owner's User Profile
Object Size (with "K" or "M" appended to indicate KBytes or MBytes)
Object auxiliary storage pool (ASP) number
Object Description (up to 50 characters; vertical bars are converted to periods)

Sample Output:

```
QSPRCLOUTQ|QRCL      |*OUTQ     |08/18/2004 13:55:43|04/17/2011 12:08:01|QSYS      |QSYS
|24576|1|SYSTEM CREATED OUTPUT QUEUE.

QSVCDRCTR |QSVCDRCTR |*OUTQ     |01/16/2002 08:23:38|04/17/2011 12:08:02|*IBM      |QSYS
|24576|1|

QEZDEBUG  |QUSRSYS   |*OUTQ     |01/22/2002 07:55:43|06/09/2011 22:00:30|*IBM      |QSYS
|24576|1|CLEANUP OUTPUT QUEUE FOR DUMPS

QEZJOBLOG |QUSRSYS   |*OUTQ     |01/22/2002 07:55:43|07/19/2011 12:59:10|*IBM      |QSYS
|475136|1|CLEANUP OUTPUT QUEUE FOR JOB LOGS

QTPPPOUTQ |QUSRSYS   |*OUTQ     |01/22/2002 07:55:43|04/17/2011 12:08:02|*IBM      |QSYS
|24576|1|TCP/IP PPP DEFAULT OUTPUT QUEUE

EOF
```

11 - Directory Listing

Returns a list of directories and file names from a specified IFS directory.

Parameter:

IFS directory name

Output: One line for each file or directory found

Sample Output: (using parameter "/var"):

```
.
..
preserve
tmp
EOF
```

16 - Job Queue Listing

Returns a list of jobs in the specified job queue.

Parameters: (separated by vertical bar)

Job queue name
Job queue library

Output: One line for each job in the queue in the format:

Job name
User profile for the job
Job number
Job type (B=batch)
Job queue priority
Submitter job name
Submitter user name
Submitter job number
Status on job queue
Date entered system (CYYMMDDHHMMSS)

Sample Output:

```
TADMINWRK |BRYAN      |010304|B| 2|QPADEV0003|BRYAN      |010246|RLS        |1050623170223
DBACKUP   |JACQUELINE|010305|B| 3|QPADEV0003|BRYAN      |010246|HLD        |1050623170513
```

20 - System Statistics

Returns one line of system statistics.

Parameters: None

Output: One line with the following format:

Number of users currently signed on
Batch jobs waiting
Batch jobs running
Batch jobs held
Batch jobs held on job queue
Batch jobs on Held job queue
Jobs in system
Percent permanent addresses
Percent temporary addresses
Percent temporary addresses system ASP (in MB)
System ASP utilization
Total aux storage
Current unprotected storage
Max unprotected storage
DB capability
Main storage size
System percent CPU (divide by 10 to get utilization in tenths)
Interval start of statistics
Interval end of statistics

Sample Output:

```
5|0|52|0|0|0|274|7|33|17549|764219|17549|1174|1213|0|0|22|1119546833|1119546893
```

21 - System Pools

Returns a line of information for each memory pool.

Parameters: None

Output: One line for each pool with the following format:

Pool number
Pool size (in KB)
Reserved size (in KB)
Maximum active jobs
Database faults – rate (in tenths) in page faults per second
Database pages – rate (in tenths) in pages/second, that database pages are brought into the pool
Non database faults – rate (in tenths) in page faults per second, of non-database page faults
Non database pages – rate (in tenths) in pages per second, that non-database pages are brought in
Active wait – rate (in tenths) in transitions per minute, of threads going from active to waiting
Wait ineligible – rate (in tenths) in transitions per minute, of threads going from wait to ineligible
Active ineligible – rate (in tenths) in transitions per minute, of threads going from active to ineligible
Pool name – numeric names for private pools associated with a subsystem; other special names:
- *MACHINE : the machine pool
- *BASE : the base system pool, which can be shared with other subsystems
- *INTERACT : the shared pool used for interactive work
- *SPOOL : the shared pool used for spooled writers
- *SHRPOOL1 - *SHRPOOL60 : a shared pool

Subsystem name – associated subsystem (blank for shared pools)
Subsystem library name
Paging option

Sample Output:

```
1|80024|43700|32767|0|0|0|3|206|0|0|*MACHINE  |          |          |*FIXED
2|168760|516|39|0|0|0|0|1693|0|0|*BASE     |          |          |*FIXED
3|13104|0|5|0|0|0|0|0|0|0|*INTERACT |          |          |*FIXED
4|256|0|5|0|0|0|0|0|0|0|*SPOOL    |          |          |*FIXED
```

22 - System CPU Utilization

Returns current system CPU utilization

Parameters: None

Output: One with the following format:

Percent CPU utilization (divide by 10 to get utilization in tenths)
Statistics interval start time
Statistics interval end time

Sample Output:

```
16|1119872296|1119872356
```

27 - Active Job Log

Returns the last 500 lines from the active job log of the specified job.

Parameters: (separated by vertical bar)

Internal Job ID in printable hexadecimal (field #4 of the code 01 output above)
Number of lines of job log output to display

Output: Job log output. The most recent lines are returned first, with the following format:

Message ID
Message type
Message severity
Message time
Sending program
Sending text

Sample Output:

```
CPC9801|01|00|1050526121203|QMHQCRTQ  |OBJECT QNMACDQ TYPE *DTAQ CREATED IN LIBRARY QTEMP.
CPI7BC4|04|00|1050526121234|QALMGR    |ALERT PROCESSING STARTED ON 05/26/05 AT 12:12:34.
CPD8C23|02|20|1050603092645|QESRSRVA  |THERE ARE ERRORS OR INFORMATION NEED TO BE SEEN ON THE JOB LOG
```

28 - QSYSOPR Inquiry Messages

Returns outstanding inquiry messages (messages needing reply) from QSYSOPR.

Parameters: None

Output: There are two lines of output for each outstanding message. The first line has the following format:

Message ID
Message key
Fully qualified job name (Jobname/User/JobID)
Message queue name
Timestamp of message
Message text

The second line contains any extra message text that exceeded the output length for the first line.

Sample Output:

```
HDV1019|93568|RBACKUP   KEN       010297|QSYSOPR   |1050623120125|Error reading
file. Continue? (Y N)


HJSD001|93728|HJSDAILY  BRYAN     010299|QSYSOPR   |1050623121644|Invalid ID on
line 340.  Continue? (Y N Q)
```

30 - Data Queue Information

Return information about the specified data queue.

Parameters: (separated by vertical bar)

Data queue name
Data queue library

Output: One line with the following format:

Data queue name
Data queue library
Message length
Key length
Sequence
Include sender ID
Force indicators
Type
Automatic reclaim
Number of messages
Maximum number of messages
Maximum entries allowed
Initial number of entries

Sample Output:

```
EVCMDQ     |EVIEW      |4104|0|F|N|N|0|0|0|1040|4048|16
```

31 - ASP Statistics

Returns statistics on Auxiliary Storage Pools

Parameters: None

Output: One line for each ASP (up to 20) with the following format:

ASP Number
Number of disk units
ASP capacity – total in MB
ASP capacity available – total in MB
ASP capacity protected – total in MB protected by mirroring or device parity
ASP capacity available protected – total in MB
ASP capacity unprotected – total in MB not protected by mirroring or device parity
ASP capacity available unprotected – total in MB
ASP system storage – storage in MB currently allocated in the ASP for system use
Overflow storage – MB of storage overflowed from the user ASP into the system ASP
Space allocated to error log – in MB
Space allocated to the machine log – in MB
Space allocated to the machine trace – in MB
Space allocated for main storage dump – in MB
Space allocated to the microcode – in MB
Storage threshold percentage – percentage used that generates a message:
    CPF0907 sent to QSYSOPR when the system ASP reaches this threshold
    CPI0953 sent to QSYSOPR when a user ASP reaches this threshold
ASP type:
    00 – the system ASP
    10 – a user ASP that does not contain libraries
    11 – a user ASP that does contain libraries

Sample Output:

```
1|1|17549|4175|0|0|17549|4175|19|0|1|55|1|139|1003|90|00
```

32 – Disk Unit Performance Statistics

Returns performance statistics on individual disk units.

Note: the performance job must be running in the agent subsystem to get valid output for this option. Use "EV400_PERF1" or "EV400_PERF2" on the node definition to activate the performance job. (See page 11.)

Parameters: None

Output: One line for each disk unit, in the format:

Disk serial number
ASP Number
Unit Number
I/Os per second (in tenths)
Reads per second (in tenths)
Writes per second (in tenths)
Disk Busy (in tenths)

Unit Control
- 0 There is no unit control value.
- 1 The disk unit is active.
- 2 The disk unit has failed.
- 3 Some other disk unit in the disk subsystem has failed.
- 4 There is a hardware failure within the disk subsystem that affects performance, but does not affect the function of the disk unit.
- 5 There is a hardware failure within the disk subsystem that does not affect the function or performance of the disk unit.
- 6 The disk unit's parity protection is being rebuilt.
- 7 The disk unit is not ready.
- 8 The disk unit is write protected.
- 9 The disk unit is busy.
- 10 The disk unit is not operational.
- 11 The disk unit has returned a status that is not recognizable by the system.
- 12 The disk unit cannot be accessed.
- 13 The disk unit is read/write protected.

Mirror Unit Protection
- 0 One mirrored unit of a mirrored pair is not active
- 1 Both units of a mirrored pair are active

Mirror Unit Reported
- 0 Mirrored unit is missing, information returned may not be current
- 1 Mirrored unit reported, information is current

Mirror Unit Status
- 1 Active
- 2 Mirrored unit being synchronized
- 3 Mirrored unit suspended

Compression Status
- 0 No compression
- 1 Compression active

Disk Protection Type
- 0 No protection
- 1 Mirrored
- 2 Part of parity protection array

Sample Output:

```
68-0DD1BE0|1|1|1|7|16|16|1|0|0|0|0|0
```

34 – Disk Unit Information

Returns information about individual disk units.

Parameters: None

Output: One line for each disk unit, in the format:

ASP Number
Disk Type
Disk Model
Disk Serial Number
Resource Name: a unique system-assigned name for the disk unit
Disk Unit Number: a unique identifier for the disk unit; mirrored disks will have the same number
Capacity: total size of the disk unit, in megabytes
Storage Available in megabytes
Storage Reserved for System in megabytes
Disk Protection Type:
0 No protection
1 Mirrored
2 Part of parity protection array

41 – List Logical Interfaces

Return information about the network interfaces

Parameter:

Interface Type (optional, default is *ALL)

Output: One line for each interface with the following format:

IP Address
Network Address
Interface Subnet Mask
Interface Name
Line Status
Interface Type
MAC Adapter Address

Sample Output:

```
127.0.0.1       |127.0.0.0       |255.0.0.0       |*LOOPBACK |ACTIVE       |NONE       |
192.168.1.113   |192.168.1.0     |255.255.255.0   |ETHLINE   |ACTIVE       |ETHERNET   |00096B6BBF83
EOF
```

42 – Network Connection Status

Return standard Netstat information about the network connections.

Parameters:

Starting Local Port (0 to 65535, default is 0)
Ending Local Port (0 to 65535, default is 65535)
Type (TCP, UDP, IPS, or *ALL, default is *ALL)
IP Version (IPV4 or IPV6, default is IPV4)

All parameters are optional, but vertical bar placeholders must be used if earlier parameters are omitted.

Output: One line for each network connection in the format:

Remote Address
Remote Port
Local Address
Local Port
Connection Type (TCP, UDP, or IPS)
User Who Performed the Bind
Idle Time (seconds)
Bytes In
Bytes Out
Connection Status
Job(s) Using the Connection (multiple jobs separated by commas)

Sample Output:

```
0.0.0.0         |    0|0.0.0.0         | 8478|TCP|QUSER     |3769403.4|      0|       0|LISTEN   |QZDASRVSD
0.0.0.0         |    0|0.0.0.0         | 8479|TCP|QUSER     |3769405.0|      0|       0|LISTEN   |QNPSERVD
127.0.0.1       |11423|127.0.0.1       | 8002|TCP|EVUSER    |    156.1|      0|1155666783|ESTABLISH|EVSTCPPROC
127.0.0.1       |37751|127.0.0.1       | 8473|TCP|QUSER     |   6885.1|   4866|   55872|ESTABLISH|QPWFSERVSO
127.0.0.1       | 8002|127.0.0.1       |11423|TCP|EVUSER    |    156.1|1155666783|       0|ESTABLISH|EVTCTLPROC
127.0.0.1       | 8473|127.0.0.1       |37751|TCP|QSECOFR   |   6885.1|  55872|    4866|ESTABLISH|QSRVMON
192.168.0.127   |54573|192.168.0.113   |   23|TCP|QTCP      |    221.7|  42338| 1186348|ESTABLISH|QPADEV000G,QTVDEVICE
0.0.0.0         |    0|192.168.0.113   |  427|TCP|QSYS      |3769414.6|      0|       0|LISTEN   |QSLPSVR
0.0.0.0         |    0|192.168.0.113   | 4800|TCP|QSYS      |3769414.6|      0|       0|LISTEN   |QSLPSVR
192.168.0.81    |49161|192.168.0.113   | 8000|TCP|EVUSER    |      5.8| 668488|296622990|ESTABLISH|EVTCTLPROC
192.168.0.98    | 4655|192.168.0.113   | 8000|TCP|EVUSER    |      5.8| 692072| 8416528|ESTABLISH|EVTCTLPROC
192.168.0.170   |54482|192.168.0.113   | 8000|TCP|EVUSER    |      5.8| 733336|2942542|ESTABLISH|EVTCTLPROC
0               |    0|0.0.0.0         |  138|UDP|QSYS      |     95.3|60298356| 2255330|UNKNOWN  |
0               |    0|0.0.0.0         |  427|UDP|QSYS      |3769420.2|      0|       0|UNKNOWN  |
0               |    0|0.0.0.0         |  427|UDP|QSYS      |3769420.7|      0|       0|UNKNOWN  |
0               |    0|192.168.0.113   |  427|UDP|QSYS      |3769420.7|      0|       0|UNKNOWN  |
EOF
```

5

Troubleshooting EView/400i

This chapter describes how to troubleshoot problems with EView/400i Discovery.

General Troubleshooting

Before you troubleshoot a particular problem you run into when installing, configuring, or using EView/400i, you should verify that your EView/400i environment is correctly installed and configured.

Correct installation and configuration of EView/400i ensures, among other things, that messages are processed correctly:

- **Message Generation**

  Messages are generated by the EView/400i system.

- **Message Interception**

  Messages are intercepted by the EView/400i policies and monitors.

Use EVSTATUS Command to Verify Status of AS/400 Agent

On the AS/400 agent, use the command EVIEW/EVSTATUS to collect the status of the several components of the EView/400i agent and their interaction with the AS/400 system. The command is called from an AS/400 terminal. The format is:

EVIEW/EVSTATUS PARM('*options*') OUTPUT(*outoption*)

where:

options One or more of the following, separated by spaces:

VER	EView/400i version information
CONF	Current distributed configuration files
JOBS	Status of EVSBS jobs
TCP	Defined TCP/IP ports and current status
DQS	Data queues status
AUD	System QAUDLVL vs. EView/400i audit options
USP	Defined user spaces
SYS	AS/400 system information
ALL	All of the above (Default)
? or **HELP**	Display help options

outoption One of:

*****	For output to a terminal
***PRINT**	For output to the user's print queue (Default)

Example call:

```
EVIEW/EVSTATUS PARM('JOBS TCP SYS') OUTPUT(*)
```

Browse the output text of this command and look for NOTE or WARNING messages that may indicate how to resolve outstanding problems. Retain a copy of the output for possible transmission to support personnel.

Specific Troubleshooting

This section explains how to solve specific problems you may encounter when using EView/400i.

Problems Distributing Configuration to the Agent

Symptom

When connecting to the AS/400 agent to distribute configuration data (message queues, desired message IDs), the Error Message window displays "Unable to open or create target file."

Solution

1. In most cases, this is caused by the User ID not having the authority to add or change objects in the EVIEW library. (To test if this is the case, try the distribution using the QSECOFR login if it is available.)
2. The user ID used for distributions must have at least the *CHANGE authority to the EVIEW library. Use the **EDTOBJAUT OBJ(EVIEW) OBJTYPE(*LIB)** command to add the user ID to the EVIEW library's authorization list.
3. The user ID must also have authority to modify the eight configuration objects within the EVIEW library. Use the **GRTOBJAUT** command to grant authority to these objects:

```
GRTOBJAUT OBJ(EVIEW/EVF020)      OBJTYPE(*FILE)   USER(user) AUT(*ALL)
GRTOBJAUT OBJ(EVIEW/EVF021)      OBJTYPE(*FILE)   USER(user) AUT(*ALL)
GRTOBJAUT OBJ(EVIEW/EVMSGQCFG)   OBJTYPE(*FILE)   USER(user) AUT(*ALL)
GRTOBJAUT OBJ(EVIEW/EVPARMS)     OBJTYPE(*FILE)   USER(user) AUT(*ALL)
GRTOBJAUT OBJ(EVIEW/EVJOBFILT)   OBJTYPE(*FILE)   USER(user) AUT(*ALL)
GRTOBJAUT OBJ(EVIEW/EVFILTRSPC)  OBJTYPE(*USRSPC) USER(user) AUT(*ALL)
GRTOBJAUT OBJ(EVIEW/EVQCFGSPC)   OBJTYPE(*USRSPC) USER(user) AUT(*ALL)
GRTOBJAUT OBJ(EVIEW/EVHSTFLSPC)  OBJTYPE(*USRSPC) USER(user) AUT(*ALL)
```

4. In rare cases, the distribution may be caused by an out of space condition on the AS/400 agent. Verify that sufficient disk space is available for the EVIEW library.

Verifying Connectivity and Agent Operation

Symptom

No apparent communication between the UCMDB probe server and the AS/400 agent.

Solution

To verify the correct operation of the server and agent components, use the following steps:

On the UCMDB probe server:

1. Start the EView/400i Task Manager. Verify that all processes are running for that agent. If a node's Command Server is running but the Master Message Server is not, this is usually due to an incorrect license key. Check the ev400mms log file for this error (step 3 below).

2. Check the status of the TCP/IP ports used to connect to the agent. For example, if the default ports (8000 and 8001) are used, issue the command **`netstat -a`** and look for ports 8000 and 8001 to have a status of "Established".

3. Check for errors in the `ev400mms.`*`as400name`*`.log` and `ev400cs.`*`as400name`*`.log` files in the \Program Files\EView Technology\EView 400\log directory.

On the AS/400 managed node:

1. Enter the **`EVSTATUS`** command as described on page 38. Look for any "Note" or "Warning" messages in the output which may indicate a problem.

2. Issue the command **`WRKACTJOB SBS(EVSBS)`**

   The following six jobs should be listed in an active (not "Message Wait") status:

```
EVACMDPROC   PGM-EVCCMD
EVCCTLPROC   PGM-EVCCTL
EVSCMDPROC   PGM-EVC050
EVSMSGPROC   PGM-EVC010
EVSTCPPROC   PGM-EVCHCI
EVTCTLPROC   PGM-EVCMSG
```

   If the QHST monitoring option was selected in the AS/400 node's configuration (EV400_MONITOR_QHST parameter is "YES"), then verify the additional job is active:

```
EVSHSTPROC   PGM-EVHSTCL
```

   If the performance monitoring option was selected in the AS/400 node's configuration (EV400_PERF1 and/or EV400_PERF2 parameter is "YES"), then verify the additional job is active:

```
EVPERFPROC   PGM-EVPERFM
```

   Check the agent message queue for any errors that may have been issued:

```
DSPMSG EVIEW/EVMSGQ
```

3. Check the agent trace files for any error output. The trace files are in the EVTRACE output queue of the EVIEW library.

4. Check the status of the TCP/IP ports used by the agent using the command **`NETSTAT *CNN`** If the OMW management server processes are connected, the ports configured in parameters EV400_AS400_MSG_PORT and EV400_AS400_CMD_PORT (8000 and 8001 by default) should show as "Established". It is normal for these two ports to also be in a "Listen" state. The port configured in parameter EV400_AS400_SERVER_PORT must be "Established" before any messages or command responses can be sent to the OMW server.

5. Check the condition of the agent data queues. The agent uses several data queues to store requests and messages. Data queue objects may become damaged due to unexpected interruption or system errors, which can cause agent jobs to fail. Issue the following commands to check the data queues:

```
ADDLIBLE EVIEW
DDQ EVIEW/EVSENDQ
DDQ EVIEW/EVAPIQ
DDQ EVIEW/EVCMDQ
DDQ EVIEW/EVMRSPQ
```

   If a data queue has been damaged, an exception message will be generated when issuing the DDQ command for that queue. If the data queue properties are displayed, verify that the maximum entry length is not zero, which is another indication of a damaged data queue.

EV/400i Agent Messages

This appendix describes all messages generated by the EV/400i jobs running on the iSeries (AS/400 agent).

EV/400i - Messages

Table A-1: Messages

Message Il	Severity	Type	Description	System Actic	User Action
EVM0001	99	Info	An invalid reply to a message	Processing continues	Contact your system administrator to reply tc the message on the AS/4
EVM007	99	Info	&1 limit reached, queue cleared. Maximum number c records were writter to the queue. This is most likely due to tl OVO Management Server not being connected to the OV OS/400 agent.	Queue is cleared. Processing continues	If this appears repeatedl contact support
EVM0015	99	Inq	***WARNING*** Processing has ende due to the possible loss of connectivity	Process stops until message is replied to	To recover enter, A=clea data queue and continue processing, B=end EVSBS subsyster and contact system administrator.
EVM0016	99	Info	&1 has been modifie	&1 data file ha been modified	Contact your system administrator
EVM0017	99	Info	Message queue &1 was not cleared	Selected message queue is not cleared when EVSBS subsystem was ended	Contact your system administrator
EVM4444	99	Info	Cannot allocate file &1	Restoration of default configuration file failed	Release the lock on &1 and try your operatic again
EVM5000	00	Info	Reply received to an outstanding inquiry message	Processing continues	Optionally, add EVM500 to the QHST message filter tab to trigger auto-acknowledgement of inquiry messages on the OMW active messag browser
EVM5555	99	Info	There is a lock on th EVIEW	Upgrade installation	To recover enter, I=ignor the message, C=cancel operation

	library	stopped	and contact system administrator

EV/400i Server Messages

This appendix describes the messages generated on the UD Server for EV/400. The message file is located in the `\local\C` subdirectory of the EV/400 installation path with filename `ev400.msg`.

EV/400 Management Server Messages

The following is a list of error messages from the EV/400 processes:

Table B-1: Socket Communication Errors

Code	Definition
EVOSOK001	%s failed calling %s, reason: %s
EVOSOK010	Unable to open %s %s socket
EVOSOK020	Unable to bind socket
EVOSOK030	Unable to set socket to non-blocking mode
EVOSOK031	Unable to set socket to blocking mode
EVOSOK040	Error on listen for socket connection
EVOSOK050	Socket connect failed, will retry momentarily
EVOSOK051	Socket connect failed, no retry will be attempted
EVOSOK070	Unable to get socket option: %s
EVOSOK071	Unable to get socket option: %s
EVOSOK080	%s failed reading MMS socket, reason: %s
EVOSOK081	Failure reading %s client UDP socket
EVOSOK082	Failure reading %s server UDP socket, number bytes returned is zero
EVOSOK083	Failure reading %s server UDP socket, entire message no sent
EVOSOK090	Failure writing to %s client UDP socket
EVOSOK092	Failure writing to %s client UDP socket, entire message not sent
EVOSOK199	Failure reading EView/Open Mainframe Message Server, reason: %s
EVOSOK200	Lost connection with EView/Open Mainframe Message Server
EVOSOK201	%s has exited due to read failure on MMS connection
EVOSOK202	%s has lost connection with the MMS
EVOSOK203	%s has exited due to loss of connection with the Comman Server
EVOSOK220	MMS failed sending command response to the Command Server
EVOSOK221	No TCP connection with Doman: %s

Table B-2: Management Platform API Errors

Code	Definition
EVOAPI001	%s can not make initial connection with management API
EVOAPI002	%s was not able fill symbol map
EVOAPI003	%s was not able fill status map
EVOAPI004	%s was not able to lock data base
EVOAPI005	%s failed trying to add a node to the management platfor %d
EVOAPI006	Current OpenView map \" % s\ " is Read-Only, exiting
EVOAPI100	API eror message: %s
EVOAPI101	%s lost connection with API: Exiting
EVOAPI200	No selected icon for %s
EVOAPI201	Only one symbol may be selected
EVOAPI205	Attempting to get the the id for %s

Table B-3: Process Initialization Errors

Code	Definition
EVOINI000	%s initialized successfully
EVOINI001	%s initialized successfully for domain %s
EVOINI010	%s started with invalid argument count
EVOINI011	Domain name must be passed in to the %s
EVOINI012	Invalided transaction program name executable used to start %s
EVOINI013	Resource name with domain extension must be passed int the %s
EVOINI020	%s encountered invalid for configuration parameter %s
EVOINI030	%s needs the %s environment variable set properly
EVOINI040	Error setting debug value, unable to find module %s in %s
EVOINI050	%s unable to open log file: exiting
EVOINI060	Required configuration parameter is missing: %s
EVOINI070	Unable to obtain memory for Status Mapping table
EVOINI071	Unable to open OV Status Map File %1$s - Reason %2$s
EVOINI072	Unable to open vpo_severity.conf file
EVOINI073	Too many entries in vpo_severity.conf file - Notify support
EVOINI074	Invalid severity range on record %1$d - Notify system administrator
EVOINI075	Warning - overlapping range in vpo_severity.conf file reco - %1$d
EVOINI076	Invalid severity in vpo_severity.conf file on record %1$d

Table B-4: Process Execution Errors

Code	Definition
EVOEXE0	%s has completed without error
EVOEXE000	%s failed calling %s
EVOEXE001	%s failed calling %s with rc: %4d
EVOEXE002	%s failed calling %s, reason: %s
EVOEXE003	Failing system command: %s
EVOEXE005	%s found bad file format %s
EVOEXE006	%s found bad file format %s, line %d
EVOEXE010	%s failed to open file %s, reason : %s
EVOEXE011	%s failed to delete file %s, reason: %s
EVOEXE012	Please check file permissions
EVOEXE015	%s failed to obtain file statistics for file %s, reason: %s
EVOEXE020	Memory allocation failure, check available memory
EVOEXE030	Unable to obtain machine name
EVOEXE031	Unable to obtain host TCP/IP address from host name: %s
EVOEXE050	Invalid selection made, please select again
EVOEXE100	%s process has exited
EVOEXE102	%s process of domain %s has exited
EVOEXE120	%s can not reach domain %s: exiting
EVOEXE130	%s received an unsuccessful return from%s: returning

Table B-5: EV/400 Client Errors

Code	Definition
EVOCLI000	Invalid -display option use: -display DisplayName
EVOCLI010	Client received 'message out of sequence' error from serve
EVOCLI030	Invalid command form specified, valid values are 1, 2 or 3

Table B-6: Active Status

Code	Definition
EVOID010	Act_stat could not open input file %s
EVOID015	Switched_pu could not open input file %s
EVOID020	Return from host is not correct in %s: returning
EVOID030	Cannot check status of Session IDs

Table B-7: Check Status

Code	Definition
EVOCS010	The name of the resource can not be found on the commar line: exiting
EVOCS011	A resource must be selected: exiting

Table B-8: Refresh

Code	Definition
EVOREF070	Starting a passive refresh

Table B-9: Master Message Server (MMS)

Code	Definition
EVOMMS001	TCP connection established from MMS to domain: %s
EVOMMS100	TCP connection lost from MMS to domain: %s
EVOMMS110	Agent version %1$d received

Table B-10: Command Server (CS)

Code	Definition
EVOCSR001	TCP connection established from CS to domain: %s
EVOCSR100	TCP connection lost from CS to domain: %s
EVOCSR110	Agent version %1$d received

Table B-11: Command and Response Director (ev400crdr)

Code	Definition
EVOLLI001	ev400crdir terminating
EVOLLI010	Unable to Initialize with OpenView process manager
EVOLLI015	Invalid value for maximum reply buffer - setting to defaul (100,000)
EVOLLI016	Maximum reply buffer too small - setting to 5000
EVOLLI020	Error initializing with OpenView Operations Managemen Server
EVOLLI025	Unable to open directory %1$s
EVOLLI030	Unable to allocate memory for reply buffer
EVOLLI040	Reply sent to %1$s
EVOLLI045	Command %1$s sent to %2$s
EVOLLI050	Domain %1$s is not configured - notify system administrator
EVOLLI060	Invalid request format for TCP request - notify system administrator
EVOLLI065	Program error-could not find client connection - notify system administrator
EVOLLI066	Program error occurred - notify system administrator
EVOLLI100	Not connected to domain: %s
EVOLLI101	Lost connection to domain: %s
EVOLLI102	Action request queue full - contact administrator
EVOLLI103	Response message exceeds buffer size
EVOLLI104	Response timeout for this action
EVOLLI105	Unable to allocate additional memory for reply
EVOLLI106	Invalid TCP Request Port specified - using default
EVOLLI107	Unable to Bind TCP Request Port

Table B-12: ev400ragt Messages

Code	Definition
EVORAG010	Unable to resolve host name %1$s
EVORAG020	Unable to open configuration file for %1$s
EVORAG030	Error retrieving configuration for %1$s
EVORAG040	Invalid port number found in configuration file for %1$s
EVORAG050	No response from %1$s or error occurred waiting for response
EVORAG100	Primary Manager switch successful
EVORAG110	Distribute All parameter is 'YES' on %1$s
EVORAG120	%1$s Management server requested for primary is not connected
EVORAG130	This server is not authorized to make this request
EVORAG900	Unknown return code %d from %1$s

Table B-13: ev400reply Messages

Code	Definition
EVOXRY010	Unable to open DISPLAY
EVOXRY020	ev400hostcmd failed

C

EView/400i Agent Jobs

This appendix describes the various jobs that run under the EVSBS subsystem on the iSeries (AS/400).

EView/400i Subsystem (EVSBS)

The jobs that execute in the EVSBS Subsystem:

1. EVACMDPROC - Establishes the TCP/IP socket for bi-directional command and response link.
2. EVCCTLPROC - Controls the processing of pre-defined API's used in command processing.
3. EVMSGQMON – Monitors message queues configured for SCAN mode monitoring.
4. EVPERFM – Gathers performance data.
5. EVSCMDPROC - Executes the command processor.
6. EVSMSGPROC - Message queue allocation and message processing.
7. EVSHSTPROC- Extracts messages at a configured time sequence from the QHST message queue depending on the message ID's added to the filter file. These messages are forwarded to the OMW server and appear in the message browser.
8. EVSRSCPROC - Monitors status changes on discovered resources at a configured time sequence.
9. EVSTCPPROC - Receives and forwards all processed messages, commands, and API instructions from a central data queue.
10. EVTCTLPROC - Controls multiple connectivity between the UCMDB probe server and the EView/400i agent.
11. EVAUDJRNL – The RCVJRNE exit which collects audit records from the QAUDJRN journal.