CA Workload Automation CA 7 Edition

Agent Cookbook

Version 12.0.00

Second Edition

This Documentation, which includes embedded help systems and electronically distributed materials (hereinafter referred to as the “Documentation”), is for your informational purposes only and is subject to change or withdrawal by CA at any time. This Documentation is proprietary information of CA and may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA.

If you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION “AS IS” WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with “Restricted Rights.” Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

Copyright © 2015 CA. All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

CA Technologies Product References

This document references the following CA Technologies products:

- CA Integrated Agent Services (CA IAS)
- CA Universal Job Management Agent (CA UJMA)
- CA Workload Automation AE (formerly CA Autosys Workload Automation)
- CA Workload Automation CA 7® Edition, (CA WA CA 7 Edition), formerly CA Workload Automation SE and CA 7® Workload Automation

Contact CA Technologies

Contact CA Support

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

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

Providing Feedback About Product Documentation

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

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

Documentation Changes

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

- DDNAME=IASCRYPT (see page 32)—Changes to this existing topic.

Contents

Chapter 1: Plan Your Enterprise Agent Network

CA WA CA 7 Edition systems and the CA Workload Automation (WA) Agents require a unique identification throughout the Workload Automation Network that uses CA WA Agents. In one workload automation network, no two CA WA CA 7 Edition systems can have the same name, and no two agents can have the same name. Thus, you must know which CA WA CA 7 Edition systems are connecting to which agents. This network requires careful setup planning so that problems do not occur as agents and scheduling managers are added, moved, or both.

This section contains the following topics:

Collect Information

Planning for your enterprise agent network consists of the following activities:

- Collect information about all the CA WA System Agents that you are defining including the following:
  - Operating environment type (Windows, UNIX, Linux, i5/OS, and so forth).
  - Agent name.
  - Agent IP Address or DNS Name.
  - Agent listening port number for communications from the CA WA CA 7 Edition tasks.
  - Is this installation a new installation or an upgrade from a UJMA agent?
- Collect information about all the CA WA CA 7 Edition tasks that are to communicate with agents including the following:
  - Scheduling Manager name.
    - We recommend that you identify the CA WA CA 7 Edition instance and the sysplex or location.
    - If you use SMF ID, and CA WA CA 7 Edition is initialized on a different LPAR, this usage is sometimes confusing to operators.
  - Scheduling Manager IP address/ DNS name.
  - Scheduling manager listening port number for communications from the agents.

- Determine whether agent plug-ins are required.

  Note: For more information about installing plug-ins, see the implementation guide for the appropriate plug-in.

  - CA Workload Automation Application Services Agent.
  - CA Workload Automation Agent for Databases.
  - CA Workload Automation Agent for Oracle E-Business Suite.
  - CA Workload Automation Agent for PeopleSoft.
  - CA Workload Automation Agent for SAP.
  - CA Workload Automation Agent for Web Services.
  - CA WA Agent for HP Nonstop.
  - CA WA Remote Execution Agent.

  Note: CA WA CA 7 Edition does not support jobs destined to the Micro-Focus agent plug-in.

- Collect information about the applications, such as environmental variables, that can execute on the agents and any related CA WA CA 7 Edition applications, jobs flows, jobs, and so forth.

Some users find it helpful to set up a table to list out the information. The table could list agent information in one section (CA WA CA 7 Edition needs this information). The table could list CA WA CA 7 Edition information in another section (the agents need this information). The appendix contains table samples. The format is only a suggestion, and you can tailor the tables to include columns you deem appropriate.

More information:

Planning Tables (see page 65)

Conversion Considerations

If you are upgrading UJMA nodes to agents, consider the following:

- Jobs executing on a UJMA node do not necessarily require a user ID and password. Most agent jobs require a user ID and password. Whether you use existing user IDs and passwords or create new ones, define them to CA WA CA 7 Edition through the AGPSWD top line command.
- Are you using the UJMA node name as the CA WA Agent name? This usage is possible when the UJMA node name is 16 bytes or less. For example, if your UJMA node name is NODEA, you can install CA WA system agent with an agent name of NODEA. This name permits the routing of CA7TOUNI or XPJOB jobs to the UJMA NODEA and agent jobs to CA WA Agent NODEA.

- Do UJMA failover procedures exist?
- Are you running adapters associated with UJMA? If so, you need CA WA Plug-ins to let jobs execute in a system like SAP or Oracle. No conversion utility is available to convert the UJMA adapter job to a CA WA Agent job.
- Define environmental variables to the CA WA Agent. If you use environmental variables on the agent or for a job, verify that you defined the appropriate environmental variables in one of the following:
  - The CA WA Agent environmental variable files
  - The parameters of the job (ENVAR keyword)
- The UJMA agent waits for all child processes to complete and returns the highest return code using the main process to the scheduling system. With the CA WA System agent, it does not wait for completion of all the child processes and sends only the return code from the main process. For scripts that spawn child processes, changes are sometimes required to the main process to wait for the child processes to complete and exit with the desired child process return code.
- With the CA WA Agent, code the full path name in the CMDNAME or SCRIPT that is to execute. If the agent is to search for the executable, set the oscomponent.lookupcommand agentparm.txt value to true. This setting causes extra overhead on the CA WA Agent to perform the lookup.

A conversion utility is available to convert UNIX or Windows XPJOB jobs to agent UNIX_JOB or NT_JOB jobs. If jobs use adapters such as SAP or Oracle, those jobs must be converted individually.

More information:

Conversion Utility (see page 42)

Chapter 2: Distributed Operating Environment

The CA WA system agents require licenses for your intended operating environments, Windows, UNIX, or both. Contact your CA Account or Sales Representative for details in pursuing this licensing.

This section contains the following topics:

Installation Choices for CA WA Agent

You have two installation choices: a silent install or interactive install. If you have multiple agents to install, consider a silent install. You can configure a properties file for each agent and then run the silent installer.

Silent Install

To perform a silent install

1. Configure the installer.properties response file.
2. Open the installer.properties response file.

   This file is available on the product CD or CA Support Online website at http://support.ca.com.
3. Edit the properties.

   Note: For more information about silent installer properties, see the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide.*
4. Remove the pound sign (#) to uncomment each property line.
5. Evaluate which options you want to activate, such as FTP, SNMP, and JMX.

   These options depend on the application jobs that are routed to these agents.
6. CA WA CA 7 Edition supports only the AES Encryption algorithms. Verify that you code STRONG_ENCRYPTION_CIPHER as AES.
7. Change the STRONG_ENCRYPTION_KEYGEN so that it contains 32 or 64 characters.
8. Save the file.
9. Run the silent installer from a command prompt in the operating environment where you are installing the CA WA Agent:

   On UNIX:

   `./setup.bin -f response file`

   On Windows:

   `setup.exe -f response file`

   response file is the fully qualified path name where the responses (installer.properties) are located.
10. Examine the following file located in the agent installation directory for any errors:

    `CA_Workload_Automation_Agent_Rvv.r_InstallLog.log`

Interactive Install

To perform an interactive install

1. Select an installation folder/path.
2. Assign an agent name.

   This name must be unique within the agent network. The maximum length of the name is 16 characters. This name must match the name field on the CA WA CA 7 Edition AGENT statement in the IASAGENT file.

3. Assign a port for communicating to CA WA CA 7 Edition. This name must match the CA WA CA 7 Edition AGENT PORT(*nnnnn*) statement in the IASAGENT file.
4. Define each CA WA CA 7 Edition task with which this agent is to communicate.

   Manager ID – This name must match the CA WA CA 7 Edition MANAGER NAME(*name*) statement in the IASAGENT file.

   Manager IP address/DNS name

   Manager Port – This name must match the CA WA CA 7 Edition AGENTRCV PORT(*nnnnn*) statement in the IASAGENT file.

   Define encryption type and key. The type must be AES, and the key length must be the full 32 or 64 characters. This key must match the CA WA CA 7 Edition CRYPTNAME(*name*) KEY(*key*) in the IASCRYPT file.

   Many more installation options are available.

   Note: For more information about agent installation options, see the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide.*
5. Install the agent using an interactive program.

   Note: For more information about installing the Agent on Windows or UNIX using an interactive program, see the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide*.

Starting, Stopping, and Status of the Agent

Start and stop the agent using commands to execute or stop the program cybAgent. Before you start the agent, verify that the agent was properly stopped.

Starting

To start the agent on the UNIX operating environment, execute one of the following:

```
./cybAgent &  (to run in the background)
```

```
./cybAgent -a
```

Note: On UNIX, run the CA WA Agent under the ROOT authority; otherwise, the ability to switch user cannot be performed. Also, if the agent executes under a specific user ID, that user ID must have access to all files and commands that can be accessed through that agent.

On Windows, various ways to execute the CA WA Agent are available.

Note: For more information about controlling the agent, see the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide*.

You can start the Windows agent in the following ways:

- Use a command prompt entering the following after changing the directory to where the agent is installed:

  ```
  cybAgent –a
  ```

- Start as a Windows Service.

The agent can also be started automatically when the Windows system is started.

Stopping

Shut down the CA WA Agent properly each time. On the UNIX system, enter the following command:

```
./cybAgent –s
```

For Windows, depending on how it was started, perform the appropriate stop function:

- If started using a command prompt, enter the following command after changing the directory to where the agent is installed:

  ```
  cybAgent –s
  ```

- If the agent executes as a Windows Service, stop the service.

You can also stop the agent by entering a CA WA CA 7 Edition command:

```
/AGENT,AGENT=agentname,FUNC=SHUTDOWN
```

Although this command stops the agent, no agent start command is available. To start the agent, go to the operating environment and enter the appropriate start command.

Status

The CA WA Agent, in the agent installation directory, has a text file named status. This file is updated with the agent status each time a start or stop of the agent is performed. This file also indicates the build information for the running agent and is useful when reporting any problems to CA Support. A sample of the status file follows:

```
CA Workload Automation Agent for Microsoft Windows Rvv.r, Build nnn
Started at: Wed Aug 18 10:06:58 20yy
OS component - 4952
```

You can also verify the agent status through the processes/services.

Note: For more information about process/service status, see the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide*.

The agentparm.txt File

When you install the agent, the installation program adds commonly configured agent parameters to the agentparm.txt file located in the agent installation directory. Other agent parameters exist that you must manually add to the agentparm.txt file. Also, changes required after the install are made directly to the agentparm.txt file.

Note: For any configuration changes to take effect, always stop and start the agent. For some agent parameters, such as the agent name and communication parameters, configure the parameters on the CA WA CA 7 Edition task.

To update or add parameters in the agentparm.txt file

1. Change to the agent installation directory.
2. Stop the agent.
3. Make a backup/copy of agentparm.txt file.
4. Open the agentparm.txt file.
5. Edit the parameters to make the required changes.
6. Save and close the agentparm.txt file.
7. Start the agent.

Set Up Security on the Agent

Setting up local security checks on the agent is optional. The agent can perform its own security checks when it receives instructions from the scheduling manager. Security rules define these checks.

Although security is mostly controlled on the CA WA CA 7 Edition side, the agent can perform its own local security checks against its security.txt file. These checks do not override the operating system security but are performed within the CA WA Agent.

Note: For more information about local security on the agent, see the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide*.

To enable this local security, in the agentparm.txt file, set parameters security.level=on and security.filename=file.txt. This file contains the local checks that the agent is to perform. The default file, security.txt, permits all access to CONTROL commands and denies all access to FTP and commands/scripts, as shown in the following:

```
c a * * *
f d * * +
x d * * +
```

c

Indicates CONTROL commands.

f

Indicates FTP access.

x

Indicates command and script execution.

a

Permits access.

d

Denies access.

The following are the masking characters:

* for zero or more characters, and when applicable to directories, it means the current directory only.

+ for zero or more characters and it applies to the current directory and all subdirectories. When dealing with a file, it implies all members within that file.

Each format for c, f, and x is a little different. The following are the formats.

Note: For more information about formats, see the *Agent Implementation Guide*.

```
c {a|d} manager_userID CONTROL command
f {a|d} FTP_userID operation path
x {a|d} manager_userID agent_userID path
```

The manager-userID is determined on CA WA CA 7 Edition and is listed as MFUser in the AFM.

Any changes to the security file require that you recycle the CA WA Agent. The security files can also be refreshed from a CA WA CA 7 Edition terminal by entering the command /AGENT,AGENT=*agentname*,FUNC=REFRESH. This command causes the agent to reset the agent security environment such that new security definitions are in place.

More information:

Initialization File Statement Modifications (see page 38)

Default User ID

A primary source for the user ID and password is from the job definition. An optional method to assign a user ID and password for all jobs executing on the agent is to assign a default user ID and password through the agentparm.txt file. The following parameters are coded:

```
oscomponent.default.user=defaultuserID
oscomponent.default.password=passwordforuserID
```

If a job definition does specify a user ID and password, the job definition overrides the agentparm.txt default specifications.

Encryption

As a minimum, use an encryption key between the agent and the CA WA CA 7 Edition task. The encryption type can be AES, AES256, or NONE (for HP Nonstop Agent). The encryption key is set on the agent when you install the agent. This key must match the key that is defined in the CA WA CA 7 Edition DD IASCRYPT: CRYPTNAME(*name*) KEY(*key*) TYPE(type).

When defining the key, code 32 hexadecimal (0-9, A-F) characters for TYPE(AES). For TYPE(AES256), code 64 hexadecimal characters. Verify that the alphabetic characters are uppercase. If the key length is not 32 or 64 characters, the encryption key does not match between CA WA CA 7 Edition and the agent. This mismatch causes communication errors.

The encryption name that you specify on the agent is coded as 0x followed by the hexadecimal characters. In the CA WA CA 7 Edition IASCRYPT definitions, create the key within the KEY( ) parameter (without the 0x prefix).

If the keys do not match between the CA WA Agent and CA WA CA 7 Edition, you can see various messages. On the agent side, the receiver or transmitter log can show a message similar to the following message.

```
<Regular:1>.CybTargetHandlerChannel.sendMessage[:550] - Error sending message to
CA7CA75: cybermation.library.communications.CybConversationException: Bad padding
```

Note: The preceding example is only the beginning part of the message.

On the CA WA CA 7 Edition side, the browse log or LOGP/S can show a message similar to the following message:

```
CIAS-09 CAIAS1100I Response code=000000nn, reason=000000nn, flags=0000
```

nn is usually 0A and 08 respectively.

Change the Encryption Key After Installation

Use the keygen utility to change the agent encryption key after the installation is complete. The keygen utility that is provided with the agent lets you encrypt a key. AES is the only supported cipher algorithm.

The command has the following format:

```
keygen 0xkey cipher destination
```

key

Specifies the 32 or 64 hexadecimal characters (0-9, A-F) that form one of the following encryption keys:

- The 16-byte key to use in the AES encryption algorithm.
- The 32-byte key to use in the AES256 encryption algorithm.

This data must match the same key that is defined on the agent side so that the scheduling system and the agent encrypt the data with the same key.

cipher

Specifies the cipher algorithm. This value must be AES.

AES is the only valid option for almost all agents when used with CA Workload Automation CA 7 Edition. The only exception is for the CA WA Agent for HP Integrity NonStop, which does not currently support encryption. If you are using the HP Integrity NonStop agent, use a value of NONE instead.

destination

(Optional) Specifies the name of a text file in the installation directory that stores the encryption key. The default file name is cryptkey.txt.

The following is a sample command:

```
keygen 0x0102030405060708090A0B0C0D0E0F00 AES
```

If you change the encryption key on the agent side, recycle the agent to make the new key effective. If the change is made on the CA WA CA 7 Edition side, the command /IAS,FUNC=RECONFIG rereads the Agent and Encryption key files to set up the new configuration.

Environmental Variables

Environmental variables are dynamic values that affect the way jobs execute on the operating environment. With CA WA Agents, these variables are set up on three levels: Agent-wide, scheduling manager (CA WA CA 7 Edition) specific, and user specific. When searching for the environmental variable, the agent looks for the user-specific environmental file and then uses the applicable manager-specific file, and lastly the agent-wide file.

To set up these files, define a .txt file with each environmental variable on a separate line, with format variable=value. On Windows, the file cannot exceed 16 KB.

Next, code the following parameters in the agentparm.txt file to invoke these environmental variables at the agent, manager, and user ID levels. The fully qualified path name to the .txt file that holds the environmental variable values follows each.

```
oscomponent.environment.variable=path
oscomponent.environment.variable_manager_mmmm=path
oscomponent.environment.variable_user_uuuu=path
```

mmmm

Indicates the manager name to which these values apply.

uuuu

Indicates the specific user ID.

For example, the agentparm.txt file may code the following:

```
oscomponent.environment.variable=C:\Program Files\CA\WA Agent Rvv.r\agentvars.txt
oscomponent.environment.variable_manager_CA71SYSPLEXA=C:\Program Files\CA\WA Agent
Rvv.r\ca71plexa.txt
oscomponent.environment.variable_manager_CA75SYSPLEXA=C:\Program Files\CA\WA Agent
Rvv.r\ca75plexa.txt
oscomponent.environment.variable_user_user1=C:\Program Files\CA\WA Agent
Rvv.r\user1vars.txt
```

Note: If you place the environmental variable files in the agent directory, and you perform a CA WA Agent upgrade, save off the variable files before performing the upgrade.

Select job types can define their own environmental variable variables using the ENVAR parameter coded in the parameters that are used to execute the job.

Applications

Verify that all applications, commands, scripts, and so forth, are available that the agent requires.

To control application executions on the agent, define a set of job classes that represent the workload. These job classes must be known on the CA WA CA 7 Edition side, because each job can define a JOBCLASS parameter in the parameter definitions. The job classes are defined to the agent using the following:

`initiators.class_`*`n`*`=`*`classname`*`,`*`number`*

n

Specifies a sequential number starting at 1 and incremented by 1.

classname

Specifies the name of the job class.

number

Represents how many “initiators” of that class are permitted to run concurrently.

Another method exists by which you can assign a job class to a job based on the Verb/Subverb passed in the message to the agent. The message is referred to as an Automated Framework Message (AFM) and has the following general format:

```
header VERB SUBVERB data
```

Set up a class name that is based on the VERB and SUBVERB. The AFMs are shown in the receiver log in the log directory where the agent is installed. You can assign a class name that is based on the VERB SUBVERB combination, and then assign to that class name a number of initiators to permit concurrent execution. The assignment of the class occurs through the agentparm.txt parameter:

`initiators.afmjobclassmap_`*`n`*`=`*`verb`*`,`*`subverb`*`,`*`name`*

where *verb* and *subverb* are the verb and subverb of the AFM, and *name* is the class name to which these types of AFMs are assigned. After you define this statement, assign the number of initiators for that class using the initiators.class_*n*. The *_n* between the AFM job class map does *not* need to correspond to the *_n* of the class parameter.

For example, one reason to assign a number is to limit the concurrent number of monitoring jobs executing. Monitors have the verb/subverb as OBJTRIG DEFINE. You can want to limit the number of concurrent monitors executing on an agent to 5, for example. Next, you can set up the initiators as the following:

```
initiators.class_1=Default,100
initiators.class_2=Monitors,5
initiators.class_3=Prod,1000
initiators.afmjobclassmap_1=OBJTRIG,DEFINE,Monitors
```

Establish a Default class for jobs and commands that are submitted without a class specification. With CA WA CA 7 Edition, commands do not have an associated job class and execute under the Default class initiator.

Note: We recommend that you set up a job class and possibly a job class map in the agentparm.txt file. Set the number of initiators to match the proxy.maxSubmitConnections value, which limits the number of simultaneous active jobs on the agent (default of four). This setting causes the waiting jobs to have a status of 'Waiting for Initiator'. Without the job class, the waiting jobs appear to hang in the ready queue. For more information, see the *CA Workload Automation Agent for Remote Execution Implementation Guide* and the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide*.

Log Files

The following topics discuss the log files.

Agent Logs

The agent creates many log files that are useful for troubleshooting or reviewing historical activity. In a standard agent installation, the agent maintains the log files in a directory named log, which resides in the agent installation directory. Two log files of particular interest are receiver.log and transmitter.log.

- receiver.log contains a record of all successfully received AFMs.
- transmitter.log contains a log of all sending activity.

The log.level parameter in the agentparm.txt file controls the amount of data written to log files.

The log levels are supported are 0 through 5, and 8:

- 0, 1, 2: log errors
- 3: add queues
- 4 ,5: add debug information
- 8: add tracing information

Note: Level 2 is adequate for general, initial testing, and level 0 is adequate for production unless problems arise requiring more details for troubleshooting.

You can change select parameters from CA WA CA 7 Edition using the /AGENT command. Generally, do not make your changes through CA WA CA 7 Edition unless requested by CA Support. One that you can change on request is the log.level property to cause the writing of more messages to the agent logs for problem resolution. Thus, assume that you are executing using log.level 0, 1, or 2, but problems start occurring. You can change the logging temporarily (until the next agent recycle) or permanently (update to the agentparm.txt file). This change permits modification without needing to locate the physical host and modifying there. Some changes do require that you recycle the CA WA Agent, but for log.level, the changes can be immediate.

The following CA WA CA 7 Edition command would change the agent logging from (example) level 2 to level 5.

```
/AGENT,AGENT=agentname,PROP=log.level=5
```

If the parameter PERSIST=YES was added, this addition causes a permanent change to the agentparm.txt file.

When reporting problems with agents, generally zip the log directory and make it available to CA Support.

To clear the CA WA Agent log files on the CA WA CA 7 Edition system, enter the following command:

```
/AGENT,AGENT=agentname,FUNC=CLRFILES
```

This command resets the log files to start at the beginning, and all previous data is cleared. If your agent has been active for a long time period, and you have encountered a problem that can be easily re-created, clearing the log files sometimes presents a smaller set of log files to CA Support and makes it easier to debug the issue.

Job Logs

Optionally, the agent creates a job log for every script or binary request that executes on the system it manages. The job log contains environment and other diagnostic information that you can use to debug failed jobs. This information is stored in separate subdirectories of the spool file directory.

The default is not to create these job logs. To activate this feature to assist in debugging failed jobs, code the following in the agentparm.txt file:

```
oscomponent.joblog=true
```

Purge these logs periodically, and you can purge them automatically with the spool files created by jobs.

Spool Files

When jobs are executed on the agent, the output is stored in a set of spool directories. You can retrieve this data using the CA WA CA 7 Edition command AGFILE TYPE=SPOOL parameter or GS (“get spool”) command line option. Each time the same job executes, the output is appended to the file when the file exists. Thus if a job failed two times and succeeded on the third execution, all three outputs of the job execution are found in the same file. The latest execution is located at the end of the file.

Because spool can consume a large amount of storage space limited only by the amount of space available in the file system, clear the spool files periodically. The method differs between a UNIX system and a Windows system.

On UNIX, the agent can clear spool files automatically. The following parameters specified in the agentparm.txt file control the frequency and criteria when the spool is cleared. The default is to disable the spool clean.

```
runnerplugin.spool.clean.enable=true
runnerplugin.spool.expire=nnt
runnerplugin.spool.sleep=nnt
```

nn

Indicates the number of the time (*t*) increments.

t

Indicates a time increment set to D (days), H (hours), or M (minutes).

If the time specified in the sleep parameter is greater than the time in the expire parameter, the expire parameter sets the sleep time. For example, to set up the agent to check the spool files every 36 hours and delete spool files that are older than 10 days, the following parameters are used:

```
runnerplugin.spool.clean.enable=true
runnerplugin.spool.expire=10D
runnerplugin.spool.sleep=36H
```

On Windows, you can clear agent spool files that are older than a specific number of days using the clearspool command. First, define the ESPAGENTDIR environment variable with the path to the agent installation directory. Enter the following command at the Windows command prompt:

```
clearspool n
```

n

Specifies the maximum number of days a spool file is maintained. The clearspool command removes all files older than *n* days.

Note: We recommend that you set the following to the same value:

- The runnerplugin.spool.expire parameter
- The number of days (*n*) for the clearspool
- The AGENTDAY parameter on the CA WA CA 7 Edition initialization file XPDEF statement

AGENTDAY specifies the number of days to retain information about the agent job on the CA7AGNT VSAM file. Once the spool file is cleared, the data on the CA7AGNT file is no longer needed.

Note: For more information about agent logs or job log, see the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide*.

FTP

The agent can run as an FTP client, FTP server, or both. You can also run the FTP workload using Secure Sockets Layer (SSL) communication.

FTP Client

You can set up the agent to run as an FTP client.

To configure the agent as an FTP client

1. Stop the agent.
2. Edit the agentparm.txt file.

   Note: For more information about configuring the agent as an FTP client, see the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide.*
3. (Optional) Configure the agent FTP client to use Secure Copy Protocol (SCP).

   Note: For more information, see the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide*.
4. Start the agent.

If local security is enabled on the agent, define FTP rules.

Note: For more information about defining FTP rules for local security on the agent, see the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide*.

Use the AGPSWD command to define each FTP user on the CA WA CA 7 Edition task.

FTP Server

You can set up the agent to run as an FTP server.

To configure the agent as an FTP server

1. Stop the agent.
2. Edit the agentparm.txt file.

   Note: For more information about configuring the agent as an FTP server, see the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide*.
3. Start the agent.

To use the agent as an FTP server, set up local security on the agent.

Each FTP user ID and password must be defined on the agent. Use the ftpusrcfg utility located in the installation directory. If you set up the agent as an FTP server during installation, you have already defined one FTP user ID and password.

On UNIX, enter the following:

`ftpusrcfg -a|-d|-m|-l userid password`

On Windows, enter the following:

`ftpusrcfg.bat -a|-d|-m|-l userid password`

-a

Adds a new user ID.

-d

Deletes the specified user ID.

-m

Changes the password for the specified user ID.

-l

Lists all entries in the ftpuser.txt file.

Note: For more information about configuring an SSL FTP, see the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide*.

SNMP Set Up

The CA WA Agent can either act as an SNMP manager or connect to an SNMP manager. This ability lets CA WA CA 7 Edition process the following SNMP job types:

- SNMP Get Attribute (SNPG_JOB)
- SNMP Set Attribute (SNPS_JOB)
- SNMP Subscribe (SNPC_JOB)
- SNMP Trap Send (SNPE_JOB)

Configuring for the SNMP options requires several agentparm.txt parameters. The agent can interface with versions 1, 2, or 3 of SNMP. If you are using version 3, specific parameters require coding. The defaults are sometimes sufficient for a number of these parameters.

Configure Agent as an SNMP Manager

You can set up the CA WA Agent as an SNMP manager to listen for or send SNMP trap messages. Activating the agent as an SNMP manager requires that you add a number of parameters to the agentparm.txt file.

Note: For more information, see the *CA WA Agent Implementation Guide*.

Configuring the SNMP trap listener requires the coding of various parameters. Some parameters are specific to SNMP version 3.

Note: For more information about these parameters, see the *CA WA Agent Implementation Guide*.

Connect the Agent to an SNMP Manager

You can connect the CA WA Agent to an SNMP manager to permit that manager to monitor and control the agent. The agent supplies the Management Information Base (MIB) file cybermation.mib. This file describes the SNMP traps and metrics used on the agent.

Note: For more information about the agentparm.txt parameters required to connect to an external SNMP manager, see the *CA WA Agent Implementation Guide.*

Other Options

A number of options are available on the CA Workload Automation Agents. The previously mentioned options are the ones for getting started with basic agent operations. All options are discussed in the *CA Workload Automation System Agent Implementation Guide*. Depending on installation needs, you can refer to this guide for some of the following options:

- UNIX User Verification through PAM
- Interactive job execution on Windows Platforms
- Wake-on-LAN – the ability to “wake up” a remote system to process work
- Connect to a JMX Console

CA WA CA 7 Edition does not currently support some selected options. Do not activate these options when operating with CA WA CA 7 Edition Version 12.0.

- Continuous monitoring
- Agent alias for cluster environments
- Selected platforms such as Micro-Focus, z/OS Agents, and OpenVMS

Chapter 3: CA WA CA 7 Edition Environment

CA WA CA 7 Edition releases starting with r11.3 include agent support. CA WA CA 7 Edition interfaces with the CA WA Agents through the CA Integrated Agent Services (CA IAS) component. This component handles the building and decoding of Automated Framework Messages (AFMs) sent to and from the agents and the TCP/IP communications interface. If you plan to use the CA WA Agent interface, verify that the SMP/E function CIASB00 is included when you install CA WA CA 7 Edition.

Note: For more information about the installation of CA WA CA 7 Edition and its components, see the *CA WA CA 7 Edition Installation Guide*.

This section contains the following topics:

Configuration for CA IAS and CA WA CA 7 Edition

The agent interface requires several files. The CA IAS component requires information about the agents with which CA WA CA 7 Edition is to communicate. CA WA CA 7 Edition requires a new VSAM file where it saves returned information about jobs sent to the agent.

CA IAS Files

The files used by the CA IAS component are described in the *CA Integrated Agent Services Implementation Guide*.

For the IASAGENT and IASCRYPT files, the files are fixed (block) with a record length 80. You can code parameters in columns 1 – 71, and a plus (+) sign indicates continued statements.

DDNAME=IASAGENT

The IASAGENT file contains the CA WA CA 7 Edition scheduling manager and CA WA Agent definitions. If an agent is not defined in this file, CA WA CA 7 Edition does not acknowledge the agent, and you can send no jobs to that agent. Because CA WA CA 7 Edition controls whether an agent can connect to the CA WA CA 7 Edition/Agent network, defining the scheduling manager in the agentparm.txt file is not sufficient.

On the other hand, assume the following:

- CA WA CA 7 Edition does have the agent information in the IASAGENT definition.
- The agent does not have any communication.manager-statements coded in the agentparm.txt file.

In this case, CA WA CA 7 Edition, on initial “handshake” with the agent, is added to the agentparm.txt file. Also, if CA WA CA 7 Edition is moved from one LPAR to another and is using the same manager name, the agent updates the agentparm.txt for the current IP address. This permits CA WA CA 7 Edition to failover to another system without needing to update any information on the agent side.

A sample member for IASAGENT is provided in the CA IAS-supplied library CIASOPTN(IASAGENT).

Note: Assume that you plan to have multiple CA 7 online systems communicate with the same set of CA WA Agents. You can simplify the configuration files by separating the IASAGENT statements into two PDS members or sequential files. Next, concatenate the files together in the CA 7 Online JCL.

- Define a unique PDS member or sequential file for each CA WA CA 7 Edition system. This member/file contains the manager-related statements (MANAGER and AGENTRCV).
- Define a common PDS member or sequential file to contain all the AGENT statements for defining the remote agents.
- Concatenate these two files under the IASAGENT DD in the CA 7 Online JCL. For example, in the instance CA71 Online JCL, have the following statements:

  ```
  //IASAGENT DD DISP=SHR,DSN=prefix.CAL2OPTN(CA71AMGR)
  //         DD DISP=SHR,DSN=prefix.CAL2OPTN(CA7AGNTS)
  ```

  For instance CA72 Online JCL, use a different member to describe the CA72 manager and use the same member as coded in the CA71 JCL to describe the agents:

  ```
  //IASAGENT DD DISP+SHR,DSN=prefix.CAL2OPTN(CA72AMGR)
  //         DD DISP=SHR,DSN=prefix.CAL2OPTN(CA7AGNTS)
  ```

Scheduling Manager Definition Statements

```
MANAGER NAME(name) RETRYINTERVAL(nnnnnn) RETRYCOUNT(nnnnnn) SLEEPTIME(nnnnnn)
```

name

(Required) Identifies the 1-16 byte scheduling manager. The name must be unique across the agent network. This name must match the communications.managerid_n field defined in the agentparm.txt file.

The other parameters define defaults for any agent statements that do not specify values explicitly. The RETRYINTERVAL specifies the number of milliseconds before CA IAS tries to resend a message. If, after the RETRYCOUNT number of times, CA IAS still cannot send a message, CA IAS puts that agent into a sleep interval based on the SLEEPTIME number of seconds. For example, if you want to wait 10 seconds between retries, perform five retries, and then pause for five minutes before trying again, code the following:

```
RETRYINTERVAL(10000)  RETRYCOUNT(5) SLEEPTIME(300)
```

The default values are to retry every 30 seconds, and after 3 times, sleep for 15 minutes before trying to communicate with the agent again:

```
RETRYINTERVAL(30000)  RETRYCOUNT(3) SLEEPTIME(900)
```

```
AGENTRCV name PORT(nnnnn)
```

name

Specifies the required name of the listening port.

PORT

Specifies the required listening port of the scheduling manager. This name must match the communications.managerport_*n* field in the agentparm.txt file.

Agent Definition Statements

```
AGENT name +
ADDRESS(ip address or DNS name) PORT(nnnnn) +
CRYPTNAME(encryption key name)  PLATFORM(type)  +
RETRYINTERVAL(nnnnnn) RETRYCOUNT(nnnnnn)  SLEEPTIME(nnnnnn)
```

name

Specifies the required 1-16 byte name of the agent. This name must match the agentname field in the agentparm.txt file.

ADDRESS

(Required) Specifies the IP address or DNS name for this agent. If you are using the DNS Name, define the name to the DNS Server that must always be available. If the agent has a static IP address, using that IP address avoids the overhead associated with performing the DNS lookup to resolve the name to an IP address.

PORT

(Required) Specifies the port on which the agent listens for incoming requests. This name must match the communications.inputport field in the agentparm.txt file.

CRYPTNAME

(Required) Specifies the name of the key found in the encryption table file (DDName=IASCRYPT). Multiple agents can use the same encryption key, and thus the CRYPTNAME can be the same for multiple agents.

The other parameters are optional. The RETRYINTERVAL specifies the number of milliseconds before CA IAS tries to resend a message. If after the RETRYCOUNT number of times CA IAS still cannot send a message, CA IAS puts that agent into a sleep interval based on the SLEEPTIME number of seconds. If these parameters are not coded on the AGENT statement, the MANAGER statement values are used to determine the intervals/count.

DDNAME=IASCRYPT

The IASCRYPT file contains the actual encryption keys for the CRYPTNAMEs mentioned in IASAGENT. This file is a separate file so that you can use external data set security to control access if necessary.

One of the most important purposes for this file is to ensure the encryption keys that are defined on the agent match the encryption keys coded in this member/file. If the encryption keys do not match between CA WA CA 7 Edition and the agent, communication is not successfully established and the agent does not execute jobs.

A sample member for IASCRYPT is provided in the CA IAS-supplied library CIASOPTN(IASCRYPT).

The following statement defines the encryption keys:

```
CRYPTNAME NAME(name) KEY(0102030405060708090A0B0C0D0E0F00) TYPE(AES|AES256|NONE)
```

CRYPTNAME

Specifies the keyword identifying the beginning of an encryption key definition.

NAME(*name*)

Specifies the name that is associated with the encryption key. The name can range from 1-16-alphanumeric characters and must begin with an alphabetic character. This name is referenced in the AGENT definitions as CRYPTNAME(*name*).

KEY(*data*)

Specifies the 32 or 64 hexadecimal characters (0-9, A-F) that form one of the following keys:

- The 16-byte key to use in the AES encryption algorithm.
- The 32-byte key to use in the AES256 encryption algorithm.

This data must match the same key that is defined on the agent side. When the data matches, the scheduling system and the agent encrypt the data with the same key.

TYPE(AES|AES256|NONE)

Specifies the type of encryption used. AES and AES256 are the only valid options for almost all agents. Although the system agent can support multiple encryption types, the only supported types are AES and AES256. The default value is AES.

The only valid exception is for the CA WA Agent for HP Integrity NonStop, which does not currently support encryption. If you are using the HP Integrity NonStop agent, use a value of NONE instead.

DDNAME=IASCKPT

The IASCKPT data set is a DIV data set backed by a VSAM Linear Data Set. The file contains the message queues for messages being sent to the agent and received from the agent before that message is passed to CA WA CA 7 Edition. The file also contains selected information about the agents (for example, statistics) and user ID/password information. The file is also referred to as the communication queue.

To allocate this file, use the sample JCL supplied in the CA IAS supplied library CIASJCL(IASCKPT).

Consider the allocated size of the data set. Because it is a DIV file, the entire file contents are loaded into storage for processing. We recommend that you start with no more than ten cylinders primary and, optionally, two cylinders secondary. The required size depends on several items:

- Each agent record requires about 500 bytes.
- Each active job requires 100 bytes + size of the AFM (AFMs vary in size according to job types and number of parameters being passed).
- Each password entry requires about 350 bytes.
- System overhead is about 1000 bytes.

For example, 25 agents, 250 average jobs, and 50 passwords, the approximate space requirement is about 8 percent of 2 cylinders (1,474,560 bytes).

Back up this file on a periodic basis. Although the backup can occur while the CA 7 Online system is active, the CA 7 Online system must be brought down when the data set is reloaded, moved, or both. In the CA IAS CIASJCL library are several members to perform backup and reload of the file:

- IASCKPBK = backup
- IASCKPRL = reload
- IASCKPRP = reload of password information only

Most of the data in the IASCKPT file is dynamic; the exception is the user ID/password information. Over time, you can add a number of these user ID/passwords to the IASCKPT file. If the agent queue information becomes corrupted, you can restore the password information without the agent queues by using a file backup and the IASCKPRP JCL.

Note: For more information about the IASAGENT, IASCRYPT, or IASCKPT files, see the *CA Integrated Agent Services Implementation Guide*.

CA WA CA 7 Edition File

The CA7AGNT file is required for the agent interface.

DDNAME=CA7AGNT

The CA7AGNT VSAM file, sometimes known as the agent information file, contains information returned from the CA Workload Automation agents when a job has been executed. The information includes last status of job information and other data required to retrieve more job information from the CA WA Agent. The information retained in this file about the agent jobs is displayed through the AGFILE command. The AGFILE command can also retrieve data from the CA WA Agent, such as spool files.

This file grows based on the number of agent jobs submitted. You can clear the file using the /DELAGNT command. The /DELAGNT command removes records from the file older than the number of days specified in the command or in the default specification of the initialization file AGENTDAY parameter. Back up and reorganize the file periodically through the normal ICDAMS utility. Because the /DELAGNT command removes records from the file, execute the command before the reorganization takes place.

Note: For more information about the AGFILE and /DELAGNT commands, see the *CA WA CA 7 Edition Command Reference Guide*.

To allocate the CA7AGNT file and initialize it with a seed record, use the CA WA CA 7 Edition SYSGEN job CA07N010 (found in the JCLLIB), which references the file under ddname DDAGENT. The job pulls in the SYSIN member AGTALLOC (also found in the JCLLIB). Depending on the number of agent jobs to execute, adjust the space allocation. Each execution of the job requires one record, variable-length, with an average size of about 400 bytes. Also, consider the number of days to retain the information about the CA7AGNT file in the space allocation. This file is active when agent jobs are executed, because the file is updated for every job execution and for various commands performed for an agent job.

Back up this file on a periodic basis because the information is required to retrieve information about executed agent jobs. In addition to the backup, reorganize the file periodically. Free space is initially 30 percent (CI) and 30 percent (CA) because record keys are tied to the job name, and the job can be executed frequently. This percentage can be adjusted based on the installation needs. The JCL procedure members in the SYSGEN JCLLIB for backup and reload are CA7AGBK and CA7AGRL, respectively.

CA 7 Online System Modifications

The following topics discuss CA 7 Online System modifications.

Set up CA WA CA 7 Edition Security Rules

Security rules are required for setting up access to the commands supporting the agent jobs. The commands protected by security include the following:

/AFM command (used by the user interface)

L2SCAFM

/AGENT command

L2SCAGNT

/DELAGNT command

L2SCDELA

/IAS command

L2SCIAS

AGFILE command

L2AGX

AGPSWD command

L2DBAPSW

JOBSTART command (used by SAP jobs with STARTMODE=N)

L2QPSTRT

LAGENT command

L2GILAGT

REPLY command (used by i5/OS jobs)

L2QPREPL

In addition to these checks, external security checks can be made for the following when the initialization SECURITY statement is coded to make these calls. Put in place appropriate rules to control the following resources:

- For job submission:

  CA7-instance.AGENTUSR.*userid.agentname*

- For commands:

  CA7-instance.AGENTMSG.*verbsubverb.agentname*

Where *userid* is the distributed operating environment user ID, *agentname* is the name of the agent to which the job or command is being submitted, and the *verbsubverb* is the command being issued. The verbs and subverbs you may want to secure include the following; not all verb/subverbs are applicable to all agents; many depend on the job type, operating environment, or both.

Verb	Subverb	Function
CONTROL	CANCEL	Cancel a job on an agent
CONTROL	CLRFILES	Clear the log files on an agent
CONTROL	EXPEDITE	Increase the priority of a job on an agent
CONTROL	GETLOGFILE	
CONTROL	GETSPOOLFILE	Retrieve spool file for a job
CONTROL	GETSPOOLFILELIST	Retrieve list of spool files (i5/OS)
CONTROL	GETRACEFILE	
CONTROL	REFRESH	Refresh the agent security rules
CONTROL	SETPROPERTY	Set a property in agentparm.txt
CONTROL	SHUTDOWN	Shut down the agent
CONTROL	STATUS	Ping an agent
DBMON	DELETE	Cancel a DB monitor job
DBTRIG	DELETE	Cancel a DB trigger job
FILETRIG	DELETE	Cancel a file trigger job
PSCMD	GETLOGFILE	
REPLY	. (period)	Reply to the i5/OS message
RUN	OACMD	Hold or cancel an Oracle job
RUN	PSCMD	Hold or cancel a PeopleSoft job
RUN	RFCR3CMD	Execute an SAP command
SQLCMD	CANCEL	Cancel a DB SQL job

Where there is a period (.), the period must be part of the rule, as it is a valid subverb.

Note: For more information, see the *CA WA CA 7 Edition Security Reference Guide*.

CA 7 Online JCL Modifications

After the preceding files for CA IAS and CA WA CA 7 Edition are defined, add the DD statements to the CA 7 Online system JCL. The ddnames are listed in the following, and these ddnames should point to the files created. The files only require a share disposition (DISP=SHR):

IASAGENT

Defines CA WA CA 7 Edition as a "manager" and lists the CA Workload Automation Agents to which this instance of CA WA CA 7 Edition can communicate.

IASCRYPT

Defines AES encryption keys to use during communications with CA Workload Automation Agents.

IASCKPT

Defines the CA IAS checkpoint data set.

CA7AGNT

Defines the agent information file for executed agent job information.

Initialization File Statement Modifications

Update the CA WA CA 7 Edition initialization file.

The SECURITY statement has several parameters that can affect the agent job interface. Determine whether you want to use external security to control jobs and commands destined for agents. If you want external security, code the following on the SECURITY statement of the initialization file:

EXTERNAL=(LOGON,AGENT,…)

Indicates to make external security calls.

AGCLASS={FACILITY|*aaaaaaaa*}

Determines the security resource class to examine.

AGUSER=(OWNER,REQ,QJCL,CA7)

Determines the mainframe user ID to pass with the job message. This value is tied to the internal agent security file “manager_userID” field.

Note: For more information, see the *CA WA CA 7 Edition Security Reference Guide*.

On the DBASE statement, you can provide the name of a default job with the DEFAULTAG parameter. This name lets you set default values according to site needs when a job is defined without specifying the values explicitly.

On the XPDEF statement is where agent job activation occurs. The two parameters are the following:

AGENTJOB=YES

Activates the agent job interface. The default is NO.

AGENTDAY=*nn*

(Optional) Specifies the number of days to retain information in the CA7AGNT file before that information is eligible for deletion. The default is 15, although you can code from 1 to 35.

Note: For more information, see the *CA WA CA 7 Edition Systems Programming Guide*.

Start CA WA CA 7 Edition

You are now ready to start CA WA CA 7 Edition with the agent interface active.

Define Agent Jobs and Schedules on CA WA CA 7 Edition

Define agent jobs to CA WA CA 7 Edition using top line commands DB.11 or AGJOB or FUNCTION 'A' from the DB menu. Another option is to convert XPJOB jobs using the conversion utility.

Note: For more information, see the *CA WA CA 7 Edition Database Maintenance Guide* and the *CA WA CA 7 Edition Interface Reference Guide*.

Create the CA WA CA 7 Edition agent job PARMLIB member. Each job type has its own set of parameters that can be coded. For example, an FTP job indicates the options to perform the FTP data transmission: what is the server, upload/download options, files, and so on. A monitor job indicates what to monitor and the thresholds to use. A DB Stored Procedure (DBSP) job indicates the stored procedure to execute. The conversion utility creates PARMLIB members for converted UNIX and NT jobs.

Note: For more information, see the *CA Integrated Agent Services User Guide*.

After the job definition and its parameters are defined, execute the LJCK command against the job defined to help ensure that no syntax errors are encountered. This command validates the parameters. With the LIST=DEBUG option on the LJCK command, you can display a simulated AFM message. The passwords on this output always show SIMULATE, and the user ID is the user ID with which the CA WA CA 7 Edition logon occurred. This user ID can be changed when the actual job is sent to the agent based on the SECURITY AGUSER parameter.

Note: For more information about LJCK, see the *CA WA CA 7 Edition Command Reference Guide*.

Use normal functions for defining schedules, triggers, and requirements for the agent jobs. In CA WA CA 7 Edition, a job is a job and the relationship of this job is defined the same way as any other job. ARF and VRM resource definitions can apply to the job.

The conversion utility maintains this information for converted UNIX and NT jobs.

Create the CA WA CA 7 Edition agent user ID and password through the AGPSWD command. CA Workload Automation agents typically require that a user ID and password are passed with the job that the agent is to execute.

Although you can refine when to apply a password to the user ID based on agent and job type, we recommend that you try and apply the “keep it simple” principle. The user ID has a four-level lookup that is used to resolve which password is used. The basic lookup involves merely a user ID and a password. In addition to a user ID, you can optionally specify the specific agent, job type, or both. For example, you can have AGENTA and AGENTB, and on these agents, the same user ID, IDZ, is used. On AGENTA, IDZ has password IDA, and on AGENTB, the password is IDB. You can define through AGPSWD command two entries: IDZ going to AGENTA uses password IDA and IDZ going to AGENTB uses password IDB. You can then further qualify that if an FTP job is defined going to AGENTA, make its password IDF. The AGPSWD field completes (user id) IDZ, (agent) AGENTA, (job type) FTP_JOB, and (password) IDF. This process can get complicated, which is why the “keep it simple” rule is suggested!

Also, through security PANEL rule L2DBAPSW, you can secure the AGPSWD command to only those users requiring the access to define or change passwords. Another difference is that to update the password, enter the old password (this requirement differs from XPSWD command). You can enter the AGPSWD through a CA WA CA 7 Edition batch interface, which can make it easier when you have restricted this access to personnel who are not familiar with the CA WA CA 7 Edition interface.

The conversion utility creates agent user ID and passwords for existing UJMA user IDs and passwords.

Note: For more information, see the *CA WA CA 7 Edition Database Maintenance Guide*.

More information:

Conversion Utility (see page 42)
AGPSWD Through a CAICCI Terminal (see page 71)

Monitor the Agent System

You can define specific job types within the CA WA CA 7 Edition task for use in monitoring the remote system on which the agent executes. These job types include the following:

CPU_MON

Monitors CPU usage of the computer where the agent is installed.

DISK_MON

Monitors available or used disk space.

IP_MON

Monitors an IP address or port status.

PROCESS_MON

Monitors the status of a process.

TEXT_MON

Monitors a text file for a text string.

EVENTLOG_MON

Monitors Windows Event Log.

SERVICE_MON

Monitors services on a Windows computer.

These job types are useful in an “operations monitoring” application (system) and within the business application itself. For instance, the operations monitoring team sometimes wants notification when the CPU percentage on the operating environment where the agent executes ever reaches 95 percent. Also the IP port 21, the standard FTP port, must always be available. The business applications monitoring can verify the following:

- The presence of a process before submitting a job that uses that process.
- That the process is down before submitting a job that prepares data for use in that process.

More information:

Application Exploitation (see page 45)

Conversion Utility

The conversion utility converts existing XPJOB jobs that run on UNIX or Windows operating environments to agent jobs. The utility provides a semi-automatic mechanism for changing your XPJOB jobs to agent jobs. The utility maintains any schedules, triggers, and requirements associated with the original XPJOB definition.

The conversion process has a number of conditions and rules so that the conversion from an XPJOB to an AGJOB completes successfully.

Note: For more information about XPJOB conversion, see the *CA WA CA 7 Edition Interface Reference Guide*.

The following is an overview of the conversion process:

- Part 1

  Make backup copies of existing XPJOB PARMLIB members.

  Generate a BTI file of LJOB and LJCL statements for each XPJOB meeting selection criteria.

- Part 2

  Generate the following:

  - The conversion file (BTI file necessary to convert XPJOB jobs to agent jobs).
  - XNODE and XPSWD cleanup file (BTI file to clean up XNODE and XPSWD entries).
  - User ID/password file (used to create agent user IDs and passwords).

- Part 3

  Convert the XPJOB jobs to agent jobs.

- Part 4

  Using the AGPSWD command, update the IASCKPT file with agent user IDs and passwords.

- Part 5

  When satisfied with conversion, clean up XNODE and XPSWD entries.

Note: For more information, see the *CA WA CA 7 Edition Interface Reference Guide*.

The conversion utility does not handle jobs going to UJMA adapters such as SAP. Convert these jobs to an appropriate job type, such as SAP_JOB.

For example, a CA7TOUNI or XPJOB executing the SUBFILE/XP EXEC sapjob has PARM*nn* coded to specify the SAP job name, report parameters, variants, and so forth. The CA WA agent job type is SAP_JOB. The parameters you may see coded include the following:

UJMA Adapter Parameter	Agent Keyword Parameter
REPORT=<report>	ARCREPORT
VARIANT=<variant>	VARIANT
EXTPGM_NAME=<command>	
EXTPGM_PARAM=<parm>	
JOBCLASS=<jobclass>	SAPJOBCLASS
JOBUSER=<userid>	SAPUSER
JOBNAME=<jobname>	SAPJOBNAME
JOBCOUNT=<instance>	(for SAP Copy) JOBCOPY
START={I|A|C}	STARTMODE
DEFINE=YES	
BDC=YES	
SPOOL=YES	PRINTSPOOLNAME
CANCEL=YES	
COPY=YES	SAP_JOB with JOBCOPY
STEPNUMBER=<step>	ABAPNAME

For CA7TOUNI or XPJOB jobs executing SUBFILE sapipkg (SAP InfoPackage), the job type is BWIP_JOB (Business Warehouse InfoPackage) and the parameters include the following:

UJMA Adapter Parameter	Agent Keyword Parameter
DATATARGET={E|C|P}	
EXTSYSTYPE={A|C}	
DATAFILENAME=<name>	
DATAFILETYPE={A|E}	
NAME=<ipkgname>	INFOPACK

UJMA Adapter Parameter	Agent Keyword Parameter
POSTDATA={Y|N}	
TECHNAME=<ipkgtechname>	
BATCHWAIT={Y|N}	
PRINTDETAIL={Y|N}	
JOBNAME=<jobname>	SAPJOBNAME
IMMEDIATE=Y	

For CA WA Agents, a PARAM control language keyword lets you specify various options.

For SAP Process Chains capchain, the job type is BWPC_JOB (Business Warehouse Process Chain) and the parameters include the following:

UJMA Adapter Parameter	Agent Keyword Parameter
IGNOREFAIL={Y|N}	
LISTJOBLOG={B|E}	
CHAINID=<chained>	CHAIN
LOGID=<logid|AUTO>	SAPUSER
START={A|R|S}	
TIMEOUTSEC=<number>	
TRACESAVE=Y	
	SAPJOBNAME (Required)

For SAP casapvar, update or create variants, no conversion is available.

Chapter 4: Application Exploitation

With the introduction of the CA WA Agents into the CA WA CA 7 Edition enterprise, the applications can control more processes (or jobs) that require execution and control the relationships between these processes. The following represents a sample application and how scheduling agent jobs from CA WA CA 7 Edition can complete the application processing. After that, other examples of agent jobs are provided.

This section contains the following topics:

Sample Application

The application processes data on the mainframe and a Windows system and eventually generates a report to a web page for the latest status of the application. Jobs are in sequence but no longer execute only on the mainframe environment. This sample illustrates the use of seamless integration across operating environments.

JOBA: A file watcher job waits for the creation of a file on WINA by some unknown process. The file is expected at 18:00 daily. This job is a scheduled job to bring in through the Schedule Scan process and should be submitted at 17:45. The file is created daily as TEST_Dmmddyy in the TESTAPP directory on drive D. JOBA is defined as a FILE_TRIGGER job, going to WINA.

```
------------------------  CA-7 Agent Job Definition  -----------------
Function: ADD       (Add, DD, Delete, Format, List, Purge, Update)
Job: JOBA

System: TESTAPP   JOBNET:           Owner:          UID: 0

Agent Job Type: FILE_TRIGGER
Agent: WINA
User:  USERA

Parmlib: &TESTJCL                   Member: JOBA     Use-Ovrd-Lib: N

EXEC: Y           Hold: N           Verify: N
DRClass:          ARFSET:           Satisfaction Lead Time:  0
WLBClass: A       WLBPRTY: 000      Clock Time: 0000
Don't Schedule Before: 00000 0000   After: 99999 0000
LTERM: MASTER     Prompt: Y         Rqmt List: Y   Rqmts Not Used: Y
```

The FILE_TRIGGER parameter member follows. Note the use of the global variable to set the current date in MMDDYY format.

```
FILENAME 'D:\TESTAPP\TEST_D&:7MM.&:7DD.&:7YY..txt' CREATE
```

When the job has been submitted, the job appears in the queues. When the job has gone to active status, the LQ,JOB=#,LIST=ALL shows the status field because the file monitor has been created. Also, note the AGJ value in the CPU SPEC/RUN value indicates that the job has reached the agent. This notification is useful if the job takes an error, and you want to know whether the error is on the CA WA CA 7 Edition side or on the agent side.

```
LQ,JOB=310,LIST=ALL
LIST=ALL CA-7#=0310                                          DATE=yy.ddd    PAGE 0001

  JOB  QUEUE CA-7 -DAY(DDD) AND TIME(HHMM)--   CPU     SCH ENTRY MSTR  JOB
  NAME  NAME JOB# DEADLINE SUB/START DUE-OUT SPEC/RUN  ID  MODE  REQ  STATUS

JOBA     ACT 0310 ddd/hhmm ddd/hhmm ddd/hhmm FTRG-AGJ  001 DEMD  000
       -------------------------- JOB INFORMATION --------------------------
       . SYSTEM NAME = TESTAPP
       . DRCLASS = **NONE**
       . AGENT: AGENTA

       ------------------------- AGENT INFORMATION -------------------------
       Job Type: FILE_TRIGGER          JobNo:
       Agent: AGENTA           Status: MONITOR Monitored for CREATE
       User: USERA

       -------------------------- PARM INFORMATION -------------------------
FILENAME 'D:\TESTAPP\TEST_D&:7MM.&:7DD.&:7YY..txt' CREATE
```

Although one can enter the AGFILE command at any time, when JOBA has completed, the AGFILE,JOB=JOBA,TYPE=INFO output appears as the following:

```
 ----------------- CA-7 Job INFO     For Agent AGENTA           ---------
Jobname: JOBA     CA7#: 0310  System: TESTAPP  SchId: 0001  Q-DtTm: yyddd hhmm


Job Type: FILE_TRIGGER
Agent: AGENTA
Host: HOST_AGENTA
FILENAME: D:\TESTAPP\TEST_Dmmddyy.txt
STATUS: File Created
```

No output (spool) is associated with FILE_TRIGGER job types.

JOBB: This job, through FTP, sends the file from the Windows system to the Mainframe system for processing. JOBB definition is FTP_JOB and also is sent to WINA for execution. WINA does have the FTP Client interface activated and thus can request the sending of the file. JOBB is triggered from JOBA.

```
------------------------  CA-7 Agent Job Definition  -----------------
Function: ADD       (Add, DD, Delete, Format, List, Purge, Update)
Job: JOBB

System: TESTAPP   JOBNET:           Owner:          UID: 0

Agent Job Type: FTP_JOB
Agent: WINA
User:  USERA

Parmlib: &TESTJCL                   Member: JOBB     Use-Ovrd-Lib: N

EXEC: Y           Hold: N           Verify: N
DRClass:          ARFSET:           Satisfaction Lead Time:  0
WLBClass: A       WLBPRTY: 000      Clock Time: 0000
Don't Schedule Before: 00000 0000   After: 99999 0000
LTERM: MASTER     Prompt: Y         Rqmt List: Y   Rqmts Not Used: Y
```

The parameter member for JOBB appears with the following statements:

```
LOCALFILENAME 'D:\TESTAPP\TEST_D&:7MM.&:7DD.&:7YY..txt’
REMOTEFILENAME “'TEST.DATA'”
SERVERADDR USCOMP99.CO.COM
/* TRANSFERCODETYPE U */
FTPFORMAT U
/* TRANSFERDIRECTION UPLOAD */
DIRECTION UPLOAD
```

After the job executes, a spool file is on the agent indicating the FTP was executed. Only part of the spool file is echoed here.

The following command is entered:

```
AGFILE,JOB=JOBB,TYPE=SPOOL
```

Entering the GS (Get Spool) line command from the command output of AGFILE,JOB=JOBB,LIST=ALL obtains the same data.

```
---------------- CA-7 Job SPOOL    For Agent WINA             ---------
Jobname: JOBB     CA7#: 0392  System: TESTSYS  SchId: 0001  Q-DtTm: yyddd hhmm
Job Type: FTP_JOB          Spool Offset: 0         Next Offset: EOF

-----------------------------------------------------------------
Output of messages for workload object JOBB.N00392/QATSTSYS.S00001D102511112/MAIN
Start date Wed mmm dd hh:mm:ss CDT yyyy
-----------------------------------------------------------------
Attempting to connect to USCOMP99.CO.COM:21
Connection established to USCOMP99.CO.COM:21
USER usera
331 Send password please.
PASS (hidden)
230 USERA is logged on.  Working directory is "USERA.".
Autodetecting...
The transfer type is ASCII
Uploading ASCII D:\TESTAPP\TEST_Dmmddyy.txt --> ARFSET.DATA

PROGRAM: QM82   MSG-INDX: 00   -- QM.8.2   --   yy.ddd / hh:mm:ss
MESSAGE: Enter Spool Offset or a command on the top line
         (Use 'PF7/PF8 to scroll thru data' )
```

JOBC, D, E and F process the data in that file to update master files, produce print files, and more. One of the outputs of these jobs is a report file. Because these jobs are “normal” CPU jobs already scheduled through CA WA CA 7 Edition, no data is shown here.

JOBG updates a form on a web page. The form indicates when the cycle completes. This job type requires the Application Services plug in to the CA WA System Agent, which permits the execution of an HTTP_JOB. (Other job types are supported such as JMS Publish, JMS Subscribe, Entity Beans and more).

```
------------------------  CA-7 Agent Job Definition  -----------------------
Function: LIST      (Add, DD, Delete, Format, List, Purge, Update)
Job: JOBG

System: TESTAPP    JOBNET:           Owner:          UID: 0

Agent Job Type: HTTP_JOB
Agent: APPSRVC_AGENT
User: USERA

Parmlib: &TESTJCL                   Member: JOBG     Use-Ovrd-Lib: N

EXEC: Y           Hold: N           Verify: N
DRClass:          ARFSET:           Satisfaction Lead Time:  0
WLBClass: A       WLBPRTY: 000      Clock Time: 0001
Don't Schedule Before: 00000 0000   After: 99999 0000
LTERM: MASTER     Prompt: Y         Rqmt List: Y   Rqmts Not Used: Y
```

The parameters for this job indicate the data is updated (POST) in the form. The parameters use CA WA CA 7 Edition global variables that are resolved at submission time.

```
INVOCATIONTYPE POST
SERVLET_URL http://application:8080
ACTION /publish/servlets/ComplServlet
PARAMETER KEYWORD(date) VALUE(&:7MM.&:7DD.&:7YY.)
PARAMETER KEYWORD(time) VALUE(&:7CTIME)
PARAMETER KEYWORD(application) VALUE(&:7SYSTEM)
```

More information:

FTP Client (see page 25)

Monitor Examples

Various monitoring job types are available with the CA WA System Agent. They monitor resources on the operating environment where the agent is executing. For selected monitors, you can monitor another IP address. You can monitor the CPU usage, a disk drive usage, an IP address, processes, and selected logs. CA WA CA 7 Edition submits these jobs. When the condition is met, the job ends when the keyword WAITMODE WAIT is coded. For immediate feedback, WAITMODE NOW tests for the condition and the job succeeds if the condition is met, or fails if the condition is not met.

Note: For WAITMODE WAIT job types, the CA 7 job number is active to that job until the condition is met. Thus, the CA 7 job number is not available for other jobs being attached to the request queue.

In the preceding FTP example, FTP uses port 21 and thus that port must be active before FTP using it for data transmission. If port 21 is not available, raise the attention of the operations staff. The following is a job that verifies port 21 is available before trying an FTP job:

```
------------------------  CA-7 Agent Job Definition  --------------------
Function: LIST      (Add, DD, Delete, Format, List, Purge, Update)
Job: AGQIP001

System: TESTAPP   JOBNET:           Owner:          UID: 0

Agent Job Type: IP_MON
Agent: WINA
User: USERA

Parmlib: &TESTJCL                   Member: AGQIP001 Use-Ovrd-Lib: N

EXEC: Y           Hold: N           Verify: N
DRClass:          ARFSET:           Satisfaction Lead Time:  0
WLBClass: A       WLBPRTY: 000      Clock Time: 0000
Don't Schedule Before: 00000 0000   After: 99999 0000
LTERM: MASTER     Prompt: Y         Rqmt List: Y   Rqmts Not Used: Y
```

The following are parameters to verify that IP port 21 is active:

```
IPADDRESS 999.99.9.999
IPPORT 21
STATUS RUNNING NOW
WAITMODE NOW
```

If the port is stopped, the job fails (back into the request queue) and the status returned indicates the fact:

```
LQ,JOB=416,LIST=ALL
LIST=ALL CA-7#=0416                                        DATE=yy.ddd    PAGE 0001

  JOB  QUEUE CA-7 -DAY(DDD) AND TIME(HHMM)--   CPU     SCH ENTRY MSTR  JOB
  NAME  NAME JOB# DEADLINE SUB/START DUE-OUT SPEC/RUN  ID  MODE  REQ  STATUS

AGQIP001 REQ 0416 ddd/hhmm ddd/hhmm ddd/hhmm OMIP-AGJ  001 DEMD  001  R-C0001
       --------------------------- JOB INFORMATION ---------------------------
       . USER HAS ACKNOWLEDGED PROMPT
       . SYSTEM NAME = TESTAPP
       . DRCLASS = **NONE**
       . AGENT: EROCAGENT

       ------------------------ REQUIREMENTS STATUS ------------------------
       _______ JOB RESTART REQUIRED

       ------------------------- AGENT INFORMATION -------------------------
       Job Type: IP_MON                JobNo:
       Agent: EROCAGENT        Status: FAILED Ping 999.99.9.999.21 is stopped
       User:
```

If the job completes, the indicated port is active, and any triggered jobs (such as the FTP job) are submitted for execution. Verify the status using the command AGFILE,JOB=AGQIP001,TYPE=INFO (or the IN line command from the AGFILE,JOB=AGQIP001,LIST=ALL command):

```
----------------- CA-7 Job INFO     For Agent EROCAGENT        ---------
Jobname: AGQIP001 CA7#: 0416  System: TESTAPP  SchId: 0001  Q-DtTm: yyddd hhmm


Job Type: IP_MON
Agent: WINA
Host: HOST-NAME-LOCATION
STATUS: Ping 99.999.9.999.21 is running
```

Perhaps there is a process that an application must have stopped before it can submit some “job” to execute. The application can help ensure that the process is stopped by setting up a process monitor job.

```
------------------------  CA-7 Agent Job Definition  --------------------
Function: LIST      (Add, DD, Delete, Format, List, Purge, Update)
Job: JOBPM

System: TESTAPP   JOBNET:           Owner:          UID: 0

Agent Job Type: PROCESS_MON
Agent: UNIXA
User:

Parmlib: &TESTJCL                   Member: JOBPM    Use-Ovrd-Lib: N

EXEC: Y           Hold: N           Verify: N
DRClass:          ARFSET:           Satisfaction Lead Time:  0
WLBClass: A       WLBPRTY: 000      Clock Time: 0001
Don't Schedule Before: 00000 0000   After: 99999 0000
LTERM: MASTER     Prompt: Y         Rqmt List: Y   Rqmts Not Used: Y
```

The control language (parameters) includes the following:

```
PROCESS abcApplication
STATUS STOPPED NOW
```

The job completes successfully when the process is not currently executing. This can be verified through the AGFILE,JOB=JOBPM,TYPE=INFO command:

```
----------------- CA-7 Job INFO     For Agent UNIXA           ---------
Jobname: JOBPM    CA7#: 1419  System: TESTAPP  SchId: 0001  Q-DtTm: yyddd hhmm


Job Type: PROCESS_MON
Agent: UNIXA
Host: HOST-UNIXA
STATUS: Process abcApplication is stopped
```

Note: For more information about monitoring jobs, see the *CA Integrated Agent Services User Guide*.

Other Examples

The following examples represent types of jobs that you can code with the CA WA Agents. Unlimited options exist, and the people responsible for the application must determine how they can exploit the new functionality offered with the CA WA Agents. This process includes exploiting the basic System Agent and the agent plug-ins such as Data Base, Application Services, SAP, Oracle, PeopleSoft and more.

When a file is created, copy the file to another location. The copying involves two jobs: one to monitor for the file creation (FILE_TRIGGER) and then a job to execute a command to copy the newly created file to another location. In this case, the job type is NT_JOB.

Define JOBFT as a FILE_TRIGGER job type to the CA Workload Automation CA 7 Edition system:

```
 ------------------------  CA-7 Agent Job Definition  -----------------------
Function: LIST      (Add, DD, Delete, Format, List, Purge, Update)
Job: JOBFT

System: TESTAPP   JOBNET:           Owner:          UID: 0

Agent Job Type: FILE_TRIGGER
Agent: WINAGENT
User:  domain\usera

Parmlib: &TESTJCL                   Member: JOBFT    Use-Ovrd-Lib: N

EXEC: Y           Hold: N           Verify: N
DRClass:          ARFSET:           Satisfaction Lead Time:  0
WLBClass: A       WLBPRTY: 000      Clock Time: 0001
Don't Schedule Before: 00000 0000   After: 99999 0000
LTERM: MASTER     Prompt: Y         Rqmt List: Y   Rqmts Not Used: Y
```

Define the following PARMLIB member:

```
FILENAME C:\env.txt CREATE
```

Define the job JOBCOPY as an NT_JOB job type to the CA WA CA 7 Edition system:

```
------------------------  CA-7 Agent Job Definition  -----------------------
Function: LIST      (Add, DD, Delete, Format, List, Purge, Update)
Job: JOBCOPY

System: TESTAPP   JOBNET:           Owner:          UID: 0

Agent Job Type: NT_JOB
Agent: WINAGENT
User:  domain\usera

Parmlib: &TESTJCL                   Member: JOBCOPY  Use-Ovrd-Lib: N

EXEC: Y           Hold: N           Verify: N
DRClass:          ARFSET:           Satisfaction Lead Time:  0
WLBClass: A       WLBPRTY: 000      Clock Time: 0001
Don't Schedule Before: 00000 0000   After: 99999 0000
LTERM: MASTER     Prompt: Y         Rqmt List: Y   Rqmts Not Used: Y
```

Define the PARMLIB member to execute the Windows command (DOS prompt window) passing the argument (ARGS) to copy the file to a different location:

```
CMDNAME C:\Windows\system32\cmd.exe
ARGS /C "copy C:\env C:\test\env"
```

Define the trigger relationship using job triggering another job (DB.2.4)

```
--------------------------  CA-7 JOB TRIGGERING  --------------------------
FUNCTION: UPD       (FORMAT,LIST,UPD)                          PAGE 0001
JOB:  JOBFT
OPT SCHID TRGD-JOB TRGID DOTM  QTM   LDTM  SBTM   *---- EXCEPTIONS ----*
 A   000  JOBCOPY        0010        0010
```

When the file is created, CA WA CA 7 Edition completes the JOBFT file trigger job and releases the JOBCOPY job. The JOBCOPY job uses the Windows command interpreter cmd.exe to copy the file to another location. To pass an argument to cmd.exe, such as the copy command in this example, enclose the argument in double quotation marks.

Note: The path to the Windows command interpreter cmd.exe depends on your Windows operating system version. On Windows NT, the path would be C:\WINNT\system32\cmd.exe. For the path your Windows operating system uses, see your Windows administrator.

Chapter 5: Troubleshooting

This section contains the following topics:

CA WA CA 7 Edition Troubleshooting

The following topics discuss troubleshooting CA WA CA 7 Edition problems.

Agent Connectivity

If the agent job is in a W-AGENT status on the LQ display, this status means that the job is queued to go to the agent, but the agent is not active. Use the following information to help determine what is the agent status.

The CA WA CA 7 Edition command LAGENT shows the connectivity status of the CA WA Agents. The short form of the command only shows the agent name, whether the connection is active, and the host name of the CA WA Agent operating environment. Filter the display by adding the AGENT={*name*|*mask*} to limit the display output. A sample of the LAGENT output follows:

```
LAGENT
AGENT=*

AGENT Name      Active  Host

WINAGENT          YES   99.999.1.999
UNIX_AGENT        NO    88.888.0.888
I5OS_DALLAS       YES   138.42.4.78
AM                NO    host.comp.com
WA_AGENT          YES   testsys.zzz.com
QAAGNT113         NO    qasystem.aaa.com
MODEL_OFFICE      NO    apptest.sss.com
TESTMVS           NO    111.1.1.11
```

The LAGENT command has two more formats:

- LIST=STATS
- LIST=ACCUM

LIST=STATS shows the current statistical information after the last time CA WA CA 7 Edition became active. This information includes details such as the last time an inbound and outbound message was sent, the number of messages sent, received, or queued. If known, this command also shows the CA WA Agent release and maintenance (build) level. This information is useful when reporting problems so that the support staff knows which level of the CA WA Agent is executing in the operating environment. A sample of LAGENT,LIST=STATS follows:

```
LAGENT,LIST=STATS
AGENT=*                LIST=STATS                                      PAGE 0001

AGENT Name: WINAGENT           Release: Rvv.r      Build: nnn
  HostName: HOST-WINDOWS
    OSText: Windows XP for x86
    Last Outbound: yy.ddd  hh:dd:ss          Last Inbound: yy.ddd  hh:mm:ss
Outbound Attempts:     13      Succeeded:     13   Failed:      0
 Inbound Attempts:     16      Succeeded:      0   Failed:      0
Outbound Messages:     13         Queued:     13   Purged:      0
 Inbound Messages:     16         Queued:     16
   Outbound Bytes:   2356  Inbound Bytes:   4288

AGENT Name: UNIX_AGENT         Release: Rvv.r      Build: nnn
  HostName: UNIX_BOSTON
    OSText: Windows XP for x86
    Last Outbound: unknown                   Last Inbound: unknown
Outbound Attempts:     15      Succeeded:      0   Failed:     16
 Inbound Attempts:      0      Succeeded:      0   Failed:      0
Outbound Messages:      0         Queued:      1   Purged:      0
 Inbound Messages:      0         Queued:      0
   Outbound Bytes:      0  Inbound Bytes:      0
```

The other display for LAGENT is LIST=ACCUM. This command displays information about the CA WA CA 7 Edition connection to the agent after the last initialization of the CA IAS Checkpoint file (IASCKPT). This information reflects the same information as LIST=STATS, but for a longer time period.

If using Domain Name Services (DNS) names, both the agent and the system on which CA WA CA 7 Edition executes must be able to resolve the names in question.

If the agent in inactive, it could be that the agent is not active in the operating environment. In that case, start the agent. In the agent path on the operating environment, a status file indicates the status, as known to the agent.

If the agent is active, there could be an issue with the encryption keys mismatching. In this case, examine and reset the encryption keys to help ensure that the keys are the same. CA WA CA 7 Edition issues a CA IAS message like the following:

```
CAIAS1100I Response code=0000000A, reason=00000008, flags=0000
```

In the agent logs, you sometimes see error messages that contain the “Bad Padding” error code. These messages sometimes relate to encryption keys.

If the agent is active, but connection to CA WA CA 7 Edition has not been established, examine the firewall between the two (the CA WA Agent and CA WA CA 7 Edition on the mainframe). The firewall sometimes blocks the communications.

Although CA WA CA 7 Edition connects to the CA WA Agent in a “persistent” state, it could be that the job has specified an invalid agent name. If the agent is not defined to CA WA CA 7 Edition, the job fails with a JCL error (R-JCLERR) on CA WA CA 7 Edition. The LQ,LIST=ALL status shows a partial message (the status field is limited to 40 bytes):

```
Status: IAS SEND_MESSAGE failed, rc=0004, Agent
```

The CA WA CA 7 Edition browse log (and log data) show the following error message:

```
SSM0-E8 JOB EROCAG01(0443) NOT SUBMITTED TO EROCAGEN1        -
         SSM0-90  SUBMIT ERROR FOR JOB EROCAG01(0443) - EC=E804F600 -
         IAS SEND_MESSAGE failed, rc=0004, Agent not found
        MCNT=001 FLAGS=34/0C/00/02/00/00/00
```

More information:

Encryption (see page 17)

Validate Correct Parameters

To validate that the parameters of a job are correct before sending it to the agent, use the LJCK command. This process uses the same processes used to build the message to the agent. The process can optionally display a simulated message to send to the agent. Use the LJCK command after the initial job setup, and any time changes are made to the parameters.

The LJCK command also resolves any global variables used in the parameter definitions. If the job is submitted, but ends up in the request queue with a JCLERR status on the CA WA CA 7 Edition, use the LJCK command to validate that CA WA CA 7 Edition can build the message correctly. If it can, then the JCLERR status sometimes results from security information.

The LJCK command displays data from the database, PARMLIB, and CA IAS. With LIST=DEBUG, the LJCK command shows the simulated AFM. Using the JOBFT from the earlier example, the LJCK output appears as the following:

```
LJCK,JOB=JOBFT,LIST=DEBUG
DSN=USERA02.DEVCA7.JCLLIB                                DATE=yy.ddd    PAGE 0001
JOB=JOBFT    MEMBER=JOBFT    JCLID=255    LIST=DEBUG

***************************** JOB DEFINITION DATA ******************************
JOBNAME : JOBFT
JOBTYPE : FILE_TRIGGER
AGENT   : WINAGENT
USER    : domain\usera

************** DATA FROM PARMLIB (WITH EXPANSION AND SUBSTITUTION)**************
     1. FILENAME C:\env.txt CREATE

***************** DATA FROM AFM (ERRORS DISPLAYED WITH <nnnnn>)*****************
AGENT EROCAGENT
FILENAME C:\env.txt CREATE

********************************** DUMP OF AFM**********************************
20100910 12583722+0400 WINAGENT * JOBFT.N00420/TESTAPP.S00000D102531258/MAIN FIL
ETRIG DEFINE Name(C:\env.txt) Type(CREATE) Password(SIMULATE) User(domain\usera)
MFUser(MASTER)

SLIN-00 REQUEST COMPLETED AT hh:mm:ss ON yy.ddd
```

JCL Error Status: Which Side?

If a job fails with a JCLERR status, the question arises who issued the error: CA WA CA 7 Edition or the agent?

```
LQ
LIST=                                                    DATE=yy.ddd    PAGE 0001

  JOB  QUEUE CA-7 -DAY(DDD) AND TIME(HHMM)--   CPU     SCH ENTRY MSTR  JOB
  NAME  NAME JOB# DEADLINE SUB/START DUE-OUT SPEC/RUN  ID  MODE  REQ  STATUS

SCWAL001 REQ 0244 223/1421 241/0357 223/1421 UNIX-AGJ  001 DEMD  001  R-JCLERR
EROCAG01 REQ 0422 253/1411  *NONE*  253/1411  WIN-     001 DEMD  002  R-JCLERR

SLIF-00 REQUEST COMPLETED AT hh:mm:ss ON yy.ddd
```

First look at the CPU SPEC/RUN column. If no AGJ is in the RUN column, it indicates that CA WA CA 7 Edition issued the error. If AGJ is present, then the job was submitted to the agent and thus the agent issued the error. If CA WA CA 7 Edition issued the error, validate the parameters using the LJCK command.

Next examine the security information. Is user ID in the job definition valid (issue AGPSWD LIST command on the user ID)? Is external security active? If so, are the proper rules defined in the security system?

If the job was submitted to the agent, examine the status field using LQ,JOB=#,LIST=ALL. Was the command found? Is there a security error on the agent side? The LQ command has only room for a limited number of bytes for the status. The agent can also return a "long status" field. The complete status information can be displayed through the following command:

```
AGFILE,JOB=jobname,TYPE=INFO
```

Exitcode (Return Code) Processing

For agent jobs, the agent, not CA WA CA 7 Edition, determines the success or failure of the job. Some job types let you code one or more EXITCODE statements to specify what return codes that the agent treats as success or failure.

By default, an exit code of 0 (zero) indicates job success, and any other code indicates job failure.

You can use the EXITCODE statement to define a single exit code or a range of exit codes as either success or failure. If you specify multiple exit codes, enter the most specific codes first followed by the ranges.

The following are some examples.

Example 1:

Exit codes 0, 2, 4 indicate success.

All others indicate failure.

```
EXITCODE 0 SUCCESS
EXITCODE 2 SUCCESS
EXITCODE 4 SUCCESS
```

Example 2:

Exit codes in the range 0 through 100 indicate success.

Exit code 19 and exit codes in the range 101-200 indicate failure.

Exit codes in the range 201 through 300 indicate success.

All other exit codes indicate failure

```
EXITCODE 19 FAILURE
EXITCODE 0-100 SUCCESS
EXITCODE 101-200 FAILURE
EXITCODE 201-300 SUCCESS
```

Example 3:

Exit codes 0, 1 indicate failure.

Exit codes 2 and above indicate success.

```
EXITCODE 0-1 FAILURE
EXITCODE 2- SUCCESS
```

Child Processes and Return Codes

When executing a command on the CA WA Agents, a command can spawn child processes that execute asynchronously from the main command. The return code from the main command is returned to CA WA CA 7 Edition on completion of the command. The child processes can continue execution even though the main command has completed.

If the return code from the child processes is to return to CA WA CA 7 Edition, the main command process must wait for the completion of the child processes. The main command process then passes back that return code as opposed to the return code from the main command. This process sometimes requires changes to the main command that is executed.

Reply for i5/OS Jobs

Agent jobs running on an i5/OS system can enter an 'Intervention required' condition and require a response before continuing to execute. This condition is indicated on a CA WA CA 7 Edition LQ display by the job having a status of W-REPLY. You can use LQ,ST=REPLY to show only jobs requiring a response. Use the following command to display the message which requires a response:

```
AGFILE,JOB=jobname,TYPE=INFO
```

To reply to the message, use the following new CA WA CA 7 Edition command:

```
REPLY,JOB=job#,MSG=message-text
```

Consider the following scenario:

An AS400_JOB is submitted to an agent. The Agent detects an 'intervention required' condition and sends a message to CA WA CA 7 Edition (for example, File already exists, reply Y/N to continue). CA WA CA 7 Edition marks the job status as W-REPLY and writes the message to the CA7AGNT VSAM file. The user enters the following command to display the job:

```
LQ,ST=REPLY
```

```
LQ
LIST=                                                    DATE=yy.324    PAGE 0001

  JOB  QUEUE CA-7 -DAY(DDD) AND TIME(HHMM)--   CPU     SCH ENTRY MSTR  JOB
  NAME  NAME JOB# DEADLINE SUB/START DUE-OUT SPEC/RUN  ID  MODE  REQ  STATUS

AS400JB1 ACT 0614 324/1005 324/0905 324/1005 OS40-AGJ  001 DEMD  000  W-REPLY
```

The user now enters the following command to display the message requiring a response:

```
AGFILE,JOB=AS400JB1,TYPE=INFO
```

```
----------------- CA-7 Job INFO     For Agent ESPAGNT113       ---------
Jobname: AS400JB1 CA7#: 0614  System: SYSTEM   SchId: 0001  Q-DtTm: 09316 1113

Job Type: AS400_JOB
Job number: 383075
Agent: ESPAGNT113
Host: USULAD10.COMP.COM
LogId/PID: 383075/ESPAGENT/CHKOBJ
Message: File already exists, reply Y/N to continue
```

The user now enters REPLY,JOB=614,M=Y so that the job can continue executing.

```
REPLY,JOB=614,MSG=Y
```

```
SPOG-00 Reply sent. Please verify job status thru the LQ command.
```

The user must now monitor the job status through the LQ command to verify that the job status changed from W-REPLY and is no longer waiting for a reply.

JOBSTART Command for Select SAP Job Types

With the SAP job types BDC_JOB and SAP_JOB, a STARTMODE parameter indicates when to release the job (start execution) in the SAP system. The values can be one of the following:

N

Do not release the job for execution.

I

Release the job immediately.

A

Release the job as soon as possible.

The job in CA Workload Automation CA 7 Edition appears in the active queue once the agent has passed it to the SAP system. The N value is like a “hold” action on the job, and the job cannot start execution until it is released. The JOBSTART command, with the following format, releases the job from the “hold” status:

```
JOBSTART,JOB=nnnn,MODE={I|A}
```

You can specify either I for immediate release or A for as soon as possible release. After the job is released, it executes, and job feedback is returned to CA Workload Automation CA 7 Edition.

Look at Spool Not Available

A job’s spool is kept on the agent spool file for a predetermined amount of time. The information for the job in CA WA CA 7 Edition in kept on the CA7AGNT VSAM file. The information for the job is necessary when retrieving the spool file when requested. When a user requests the agent job’s spool data, CA WA CA 7 Edition looks up the needed information in the CA7AGNT file and sends a request to the agent. If the agent cannot find the spool file for the job, the agent returns a “spool not available” message. This message can indicate that the spool is already cleared from the agent files.

Look at Spool Beyond EOF

Also, CA WA CA 7 Edition lets a user specify the offset to start at when retrieving the spool data. If the CA WA CA 7 Edition user entered a high number as the beginning offset, the “beyond EOF” message is sometimes received. In this case, enter a lower number or default to offset 0, which starts at the beginning of the spool file.

Agent Troubleshooting

The *CA WA Agent Implementation Guide* has a troubleshooting chapter that discusses many items. Refer to that document for details not covered here.

Bad Padding

If a mismatch between the encryption key used on the CA WA Agent and CA WA CA 7 Edition exists, the agent log file sometimes shows the “bad padding” messages when receiving messages from CA WA CA 7 Edition. In this case, examine the encryption key on CA WA CA 7 Edition. Reset the encryption key of the agent using the keygen utility so that it matches the encryption key that CA WA CA 7 Edition uses.

If the agent is connected to by more than one CA WA CA 7 Edition (or other scheduling managers such as CA Workload Automation AE), it is important that all scheduling managers and the agent have the same encryption key. Because the encryption key on the agent is encrypted, careful reset of the encryption keys must be performed.

Slow Responses

Slow responses between the agent and CA WA CA 7 Edition can result from the TCP/IP network. Evaluate the network configuration so that you have an efficient path between CA WA CA 7 Edition scheduling manager and the agent. Also, verify that the tasks are given sufficient executing priorities. Although most messages exchanged are relatively small, allow for the ability to transmit a 24-KB message between the agent and CA WA CA 7 Edition.

Domain Name Services (DNS) or IP Addresses

You can code either the TCP/IP numeric address or a Domain Name Services (DNS) name for the host name. If you code a DNS name, the DNS lookup must be executed for each connection. The connection between the agent and CA WA CA 7 Edition are not persistent; a connection is established for each message sent from CA WA CA 7 Edition to the agent or the reverse. Thus, the DNS name must be recognizable on both the CA WA CA 7 Edition side and the agent side. The DNS lookup is an “expensive” service, and you can realize some performance improvement when the TCP/IP address is coded as opposed to the DNS name.

Appendix A: Additional Information

This section contains the following topics:

Planning Tables

The following is agent information (used to build CA WA CA 7 Edition AGENTDEF and some select parameters for CA WA Agent):

Agent Name	Operating Environment	Character Set	IP Address/ DNS Name	Listening Port	AES Encryption Key	Retry Interval (ms) Count and Sleep
agentone	NT	ASCII	123.45.678.9	7520	0x1234567890 ABCDEF012345 67890ABCDEF	60000/5/300
prodagent	UNIX	ASCII	One.com	7520	0x0011223344 5566778899AA BBCCDDEEFF	12000/3/600

The following is CA WA CA 7 Edition information (used to complete agent definition about managers):

CA WA CA 7 Edition Name	IP Address/ DNS Name	Listening Port
CA71TEST	987.65.4.321	47010
CA71SYSPLEXA	PROD.COMPANY.COM	47010
CA72SYSPLEXA	PROD.COMPANY.COM	47011

Matching Agent Information with CA WA CA 7 Edition Information

The following table shows the parameters in the agentparm.txt file and the correlating parameters in the CA WA CA 7 Edition/IAS files. If these parameters do not match, communications are not established correctly.

agentparm.txt	**IASAGENT statement**	**keyword**
agentname	AGENT	Positional, immediately following AGENT
communication.inputport	AGENT	PORT(*nnnnn*)
IP address or DNS name of the agent (not referenced in agentparm.txt)	AGENT	ADDRESS(*xxxxxxxxxxxxxxxx*)
communication.managerid	MANAGER	NAME(*xxxxxxxx*)
communication.managerport	AGENTRCV	PORT(*nnnnn*)
communication.manageraddress	n/a	IP address or DNS name of the manager

agentparm.txt	**IASCRYPT statement**	**keyword**
Encryption key defined during agent installation (not listed in agentparm.txt)	CRYPTNAME	KEY(*xxxxxxxxxxxxxxxxxxxxxxxxx*)

Sample Input Parameter Files

The following are sample files defined to CA 7 Online and to the CA WA Agent. The items shown in *italics* are items that must be coordinated between the CA WA CA 7 Edition members and the agentparm.txt file. These items are only samples and do not show any more data than necessary.

IASAGENT and IASCRYPT

The following is a sample IASAGENT file:

```
MANAGER  +
         NAME(CA7CA75)

 AGENTRCV CA75 PORT(4701)

 COMMQ DDNAME(IASCKPT)

AGENT USER023ESP +
      PLATFORM(NT) ASCII CRYPTNAME(KEY6) +
      RETRYINTERVAL(4000) RETRYCOUNT(5) SLEEPTIME(300) +
            ADDRESS(15.155.0.208) PORT(7520)
```

The following is a sample IASCRYPT file:

```
CRYPTNAME NAME(KEY6) KEY(0102030405060708090A0B0C0D0E0F00) TYPE(AES)
```

Agentparm.txt

The following is a sample agentparm.txt file:

```
#
# Agent settings for nt-x86-32
#
# Log
#
# log.archive settings:
#  0 - archive with time extension
#  1 - append ".archive.log"
#  2 - delete previous log
#  3 - keep writing to the same log file
log.level=0
log.archive=1
log.maxsize=1M

#
# Agent name
#
agentname=USER023ESP
```

```
#
# Communications
#
communication.inputport=7520

communication.managerid_1=CA7CA75
communication.manageraddress_1=USCOMP99
communication.managerport_1=4701
communication.monitorobject_1=CA7CA75/AGENTMON1.0/MAIN

#
# Security
#
security.filename=C:/Program Files/CA/WA Agent Rvv.r/security.txt
security.level=off
security.cryptkey=C:/Program Files/CA/WA Agent Rvv.r/cryptkey.txt
```

(More of the agentparm.txt member follows but has been truncated here.)

List of Supported Agent Job Types

The following table lists all the agent job types supported in CA WA CA 7 Edition. This table also lists the Data Base menu option (DB.A.-) and the specific job type coded in the job definition panel. The alias column shows the short name that appears in displays such as LJOB MAINID column or LQ CPU SPEC column.

DB Panel	Agent Job Type	Alias	Agent Job Description
DB.A.B	UNIX_JOB	UNIX	Generic UNIX
DB.A.C	NT_JOB	WIN	Microsoft Windows
DB.A.D	FILE_TRIGGER	FTRG	File Trigger
DB.A.E	Not applicable		FTP Jobs Menu
DB.A.E.A	FTP_JOB	FTP	FTP
DB.A.E.B	SCP_JOB	SCP	Secure Copy
DB.A.E.C	SFTP_JOB	SFTP	Secure File Transfer
DB.A.F	Not applicable		SAP Jobs Menu
DB.A.F.A	BDC_JOB	BDC	SAP Batch Input Session
DB.A.F.B	BWIP_JOB	BWIP	SAP Business Warehouse InfoPackage
DB.A.F.C	BWPC_JOB	BWPC	SAP Business Warehouse Process Chain

DB Panel	Agent Job Type	Alias	Agent Job Description
DB.A.F.D	SAP_JOB	SAP	SAP Generic
DB.A.F.E	SAPA_JOB	SAPA	SAP Archive
DB.A.F.F	SAPE_JOB	SAPE	SAP Event Monitor
DB.A.F.G	SAPM_JOB	SAPM	SAP Process Monitor
DB.A.G	PS_JOB	PS	PeopleSoft
DB.A.H	Not applicable		Oracle Menu
DB.A.H.A	OA_JOB	ORAJ	Oracle Request
DB.A.H.B	OAC_JOB	ORAC	Oracle Copy
DB.A.I	Not applicable		Object Monitor Menu
DB.A.I.A	CPU_MON	OMCP	CPU Monitor
DB.A.I.B	DISK_MON	OMDK	Disk Monitor
DB.A.I.C	IP_MON	OMIP	IP Monitor
DB.A.I.D	PROCESS_MON	OMPM	Process Monitor
DB.A.I.E	TEXT_MON	OMTF	Text File Monitor
DB.A.I.F	EVENTLOG_MON	OMEL	Event Log Monitor
DB.A.I.G	SERVICE_MON	OMSM	Service Monitor
DB.A.J	Not applicable		Database Jobs Menu
DB.A.J.A	SQL_JOB	SQL	Database SQL
DB.A.J.B	DBSP_JOB	DBSP	Database Stored Procedure
DB.A.J.C	DB_MON	DBMN	Database Monitor
DB.A.J.D	DB_TRIG	DBTR	Database Trigger
DB.A.K	AS400_JOB	OS40	AS400/OS400
DB.A.L	Not applicable		Java Jobs Menu
DB.A.L.A	JMSP_JOB	JMSP	J2EE JMS Publish
DB.A.L.B	JMSS_JOB	JMSS	J2EE JMS Subscribe
DB.A.L.C	EJBE_JOB	EJBE	J2EE Entity Bean
DB.A.L.D	HTTP_JOB	HTTP	J2EE HTTP/Servlet
DB.A.L.E	POJO_JOB	POJO	J2EE POJO
DB.A.L.F	RMI_JOB	RMI	J2EE RMI
DB.A.L.G	EJB_JOB	EJBS	J2EE Session Bean

DB Panel	Agent Job Type	Alias	Agent Job Description
DB.A.L.H	JMXB_JOB	JMXB	JMX-Mbean Attribute Get
DB.A.L.I	JMXA_JOB	JMXA	JMX-Mbean Attribute Set
DB.A.L.J	JMXO_JOB	JMXO	JMX-Mbean Operation
DB.A.L.K	JMXS_JOB	JMSX	JMX-Mbean Subscribe
DB.A.L.L	JMXN_JOB	JMXN	JMX-Mbean Create Instance
DB.A.L.M	JMXR_JOB	JMXR	JMX-Mbean Remove Instance
DB.A.M	Not applicable		SNMP Jobs Menu
DB.A.M.A	SNPG_JOB	SNPG	SNMP Get Attribute
DB.A.M.B	SNPS_JOB	SNPS	SNMP Set Attribute
DB.A.M.C	SNPC_JOB	SNPC	SNMP Subscribe
DB.A.M.D	SNPE_JOB	SNPE	SNMP TrapSend
DB.A.N	WEB_SERV	WEBS	Web Services
DB.A.O	WOL_JOB	WOL	Wake-On-LAN
DB.A.P	PROXY_JOB		Remote Execution
DB.A.Q	NONSTOP_JOB	NSTP	HP Integrity NonStop

AGPSWD Through a CAICCI Terminal

The following JCL example shows add, update, and delete of an AGPSWD through a CAICCI terminal interface. You can use a batch terminal (BTI) or a TCP/IP terminal by adjusting the JCL execute and DD statements accordingly.

This JCL adds a password AbCdEfGhIjK for a user ID USERTEST. The JCL then updates the password to be TuVwXyZ for the USERTEST user ID. Finally it deletes the USERTEST. Remember that the user ID and passwords are case-sensitive!

```
//STEP1   EXEC  PGM=CAL2X2WB,PARM='CCISENF,CA77'
//STEPLIB  DD  DISP=SHR,DSN=cai.CA7.CAL2LOAD
//SYSPRINT DD  SYSOUT=*
//ERRORS   DD  SYSOUT=*
//SYSUDUMP DD  SYSOUT=*
//SYSIN    DD   *
/LOGON user,pswd
DB
AGPSWD
ADD,AG1=USERTEST,AGNEWPW=AbCdEfGhIjK,AGVERPW=AbCdEfGhIjK
AGPSWD
UPDATE,AG1=USERTEST,AGOLDPW=AbCdEfGhIjK,
AGNEWPW=TuVwXyZ,AGVERPW=TuVwXyZ
AGPSWD
DELETE,AG1=USERTEST
/LOGOFF
/*
```

The CA WA CA 7 Edition responses for this test appear as (only a subset of the commands is shown):

```
  10235  152036    DB
SDM0-00 ENTER CA-7 TRANSACTION
  10235  152036    AGPSWD
SMX3-00 Enter input data
  10235  152036    ADD,AG1=USERTEST,AGNEWPW=AbCdEfGhIjK,AGVERPW=AbCdEfGhIjK
SMX3-01 Password added successfully
  10235  152036    AGPSWD
SMX3-00 Enter input data
  10235  152036    UPDATE,AG1=USERTEST,AGOLDPW=AbCdEfGhIjK,
                   AGNEWPW=TuVwXyZ,AGVERPW=TuVwXyZ
SMX3-04 Password updated successfully
  10235  152036    AGPSWD
SMX3-00 Enter input data
  10235  152036    DELETE,AG1=USERTEST
SMX3-02 Password deleted successfully
```