

CA Workload Automation AE

Overview Guide

r11.3



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 may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA. This Documentation is confidential and proprietary information of CA and may not be disclosed by you or used for any purpose other than as may be permitted in (i) a separate agreement between you and CA governing your use of the CA software to which the Documentation relates; or (ii) a separate confidentiality agreement between you and CA.

Notwithstanding the foregoing, 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 © 2010 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 Access Control
- CA AutoSys Workload Automation Connect Option (CA AutoSys WA Connect Option)
- CA Embedded Entitlements Manager (CA EEM)
- CA Job Management Option
- CA Jobtrac™ Job Management (CA Jobtrac JM)
- CA Network and Systems Management (CA NSM)
- CA NSM Event Management
- CA NSM Management Command Center (CA NSM MCC)
- CA Scheduler® Job Management (CA Scheduler JM)
- CA Service Desk
- CA Spectrum Automation Manager (formerly named CA DCA Manager)
- CA Universal Job Management Agent (CA UJMA)
- CA Workload Automation AE (formerly named CA AutoSys Workload Automation)
- CA Workload Automation Agent for UNIX (CA WA Agent for UNIX)
- CA Workload Automation Agent for Linux (CA WA Agent for Linux)
- CA Workload Automation Agent for Windows (CA WA Agent for Windows)
- CA Workload Automation Agent for i5/OS (CA WA Agent for i5/OS)
- CA Workload Automation Agent for Application Services (CA WA Agent for Application Services)
- CA Workload Automation Agent for Web Services (CA WA Agent for Web Services)
- CA Workload Automation Agent for Databases (CA WA Agent for Databases)
- CA Workload Automation Agent for SAP (CA WA Agent for SAP)
- CA Workload Automation Agent for PeopleSoft (CA WA Agent for PeopleSoft)
- CA Workload Automation Agent for Oracle E-Business Suite (CA WA Agent for Oracle E-Business Suite)
- CA Workload Automation Agent for z/OS (CA WA Agent for z/OS)
- CA Workload Automation EE (formerly named CA ESP Workload Automation)
- CA Workload Automation SE (formerly named CA 7 Workload Automation)

- CA Workload Control Center (CA WCC)
- CA Desktop and Server Management (CA DSM)

Contact CA Technologies

Contact CA Support

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

- 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

Provide Feedback

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

If you would like 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>.

Contents

Chapter 1: Reliable Workload Automation 7

Overview	7
The Purpose of This Guide	7
Approach to Enterprise-Wide Workload Automation	8
Distinctive Features	9
More Information	9

Chapter 2: Architecture 11

Instance	11
CA Workload Automation AE Architecture	12
Event Server	13
Application Server	14
Scheduler	14
Agents and Agent Plug-ins	15
Legacy Agent Replaced by CA Workload Automation Agent	16
Client	16
CA Workload Automation AE SDK	16
Interface Components	17
Centralized Control	17

Chapter 3: Jobs 19

Scheduling Flexibility	19
Job Types	20

Chapter 4: Security 23

Overview	23
System-Level Security	23
Encryption and FIPS 140-2 Compliance	24
Native Security	25
External Security	26
Modifying Your Security After Installation	26

Chapter 5: Key Features 27

Fault-Tolerant Operation and Workload Balancing	27
High Availability: Shadow Scheduler	28

High Availability and Dual Event Servers: Tie-breaker Scheduler	28
Cross-Instance and Cross-Platform Scheduling	29
CA Workload Automation AE Cross-Instance Job Dependencies	29
Cross-Platform Scheduling	29
Cross-Instance Job Dependencies with CA Workload Automation EE	31
Real and Virtual Resources	31
Integration with Other CA Products	32
Monitoring and Reporting Jobs	32

Index	35
--------------	-----------

Chapter 1: Reliable Workload Automation

This section contains the following topics:

[Overview](#) (see page 7)

[The Purpose of This Guide](#) (see page 7)

[Approach to Enterprise-Wide Workload Automation](#) (see page 8)

[Distinctive Features](#) (see page 9)

[More Information](#) (see page 9)

Overview

CA Workload Automation AE is an industry-leading workload automation system that provides unparalleled reliability to support your mission-critical applications, enterprise-wide. Designed for distributed environments, CA Workload Automation AE delivers event-driven scheduling, centralized real-time monitoring and programmable error recovery—providing reliability and scalability to your production environment.

In addition, CA Workload Automation AE easily accommodates the massive processing of jobs required by eBusiness, providing the most rapid and reliable Internet job management for all your business needs.

The Purpose of This Guide

This guide introduces you to CA Workload Automation AE. By the time you finish reading this guide, you will have an overview of the wide scope of the product and its usability will be familiar to you. It is important to us that you feel comfortable with CA Workload Automation AE before you begin to use it.

Note: The term *Windows* refers to any Microsoft Windows operating system supported by CA Workload Automation AE unless otherwise noted. For information about which specific Microsoft operating systems CA Workload Automation AE supports, see the *Release Notes*.

Approach to Enterprise-Wide Workload Automation

While the enterprise is constantly expanding to accommodate new business initiatives, including eBusiness and decision support, many organizations have deployed mission-critical applications across a wide array of mainframe and open system environments. The goal is to achieve cost-effective information processing while meeting business requirements. This is possible only with an emphasis on optimized workload automation—the effective administration and execution of jobs using centralized process control.

CA Workload Automation AE offers an advanced solution to enterprise-wide workload automation by encompassing three key initiatives:

Cross-Platform Management

Enables cross-product and cross-platform integration with central management to optimize the solution by platform.

Integrated Management

Integrates with the leading Enterprise Management Solutions from CA and third-party products, such as ERP applications from SAP, PeopleSoft, Oracle, and other leading ERP vendors. The combined scope of CA workload automation solutions provides a total enterprise solution, which is one of the best in breadth and depth in the industry.

Superior Administration and Visualization

Bases the most powerful technology for managing jobs across the entire range of platforms, integrating with all the existing technology, including applications, databases, data movement solutions, backup and recovery systems, and other scheduling managers. CA Workload Automation AE lets you view and manage all relevant jobs and their interdependencies from a single point and presents the various aspects of workload automation through a sophisticated visualization user interface, CA WCC.

Distinctive Features

The following are distinctive features of CA Workload Automation AE:

Increased Performance

Improves performance using multi-threaded processing to enable more jobs to run in parallel.

Enterprise-wide Cross-Platform Workload Automation

Simplifies cross-platform workload automation through direct integration communication to CA Workload Automation solutions.

Web-based User Interface

Provides a single point of control and administration of jobs and reduces IT learning and operational costs.

Multi-Platform Support

Helps customers manage their enterprise-wide processes on various platforms.

Unparalleled Reliability and Scalability

Scales tens of thousands of jobs easily over hundreds of multi-platform computers, while maintaining its hallmark reliability.

More Information

After reading this *Overview Guide*, you can refer to all the resources available to you for additional information. Your product DVD contains instructional documents that showcase your software and provide detailed explanations about the product's comprehensive, feature-rich components.

Chapter 2: Architecture

This section contains the following topics:

[Instance](#) (see page 11)

[CA Workload Automation AE Architecture](#) (see page 12)

[Centralized Control](#) (see page 17)

Instance

A CA Workload Automation AE *instance* is a licensed version of CA Workload Automation AE software running as a server with one or more clients or agents. Clients and agents can run on a single computer or on multiple computers. An instance uses its own scheduler, application server, and event server and operates independently of other instances.

The instance ID (an uppercase, three-character alphanumeric name) that is referenced by the AUTOSERV environment variable identifies a CA Workload Automation AE server installation on a particular computer. The default instance ID is ACE. However, you can specify a different instance ID only during installation.

Multiple instances can run on the same computer, but they must have different instance IDs. For example, you can have one instance for production and another for development. Multiple instances can run on the same computer using a single copy of the binaries, and can schedule jobs on the same computers without interfering or affecting other instances.

CA Workload Automation AE Architecture

The CA Workload Automation AE architecture is designed to deliver workload automation essentials for mission-critical applications, enterprise-wide. Its event-driven, tiered architecture comprises the following system components:

Event server (database)

Stores all the objects that are used by CA Workload Automation AE.

Application server

Acts as a communication interface between the event server and the client utilities.

Scheduler

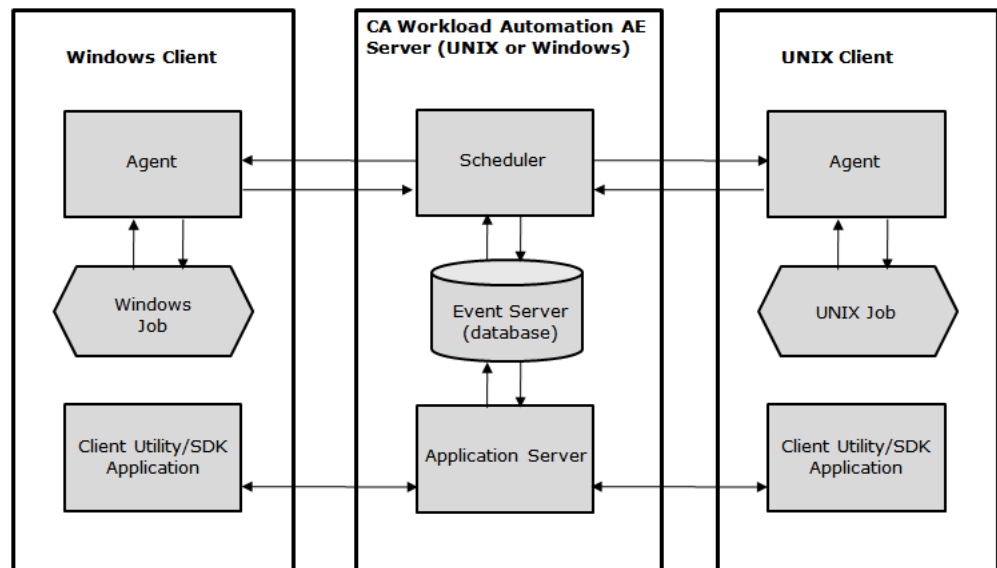
Interprets events and, based on job definitions, initiates actions through the agent. This is the engine of CA Workload Automation AE.

Agent

Performs tasks and sends the resulting job status back to the scheduler.

Note: For more information about the CA Workload Automation AE system components and how they interact with each other, see the *UNIX Implementation Guide* or *Windows Implementation Guide*.

The following illustration shows the system components in a basic configuration, and displays the communication paths between them:



Event Server

The *event server* (database) stores all the objects that are used by CA Workload Automation AE. The job, machine, and calendar object definitions comprise a subset of the data contained in the event server as do job events. The application server manages the creation, update, and deletion of the CA Workload Automation AE objects in the event server. The scheduler polls the event server for job events and fetches the corresponding object definitions that are referenced by the event when necessary.

CA Workload Automation AE supports Oracle, Sybase, and Microsoft SQL Server databases. Only the scheduler and the application server processes interface directly with the database. Therefore, these processes require a vendor database client installation to access the database. All other CA Workload Automation AE processes interface with the application server and do not require database client installations. The scheduler and the application server interact with the database using vendor-specific native code libraries. They do not use Open Database Connectivity (ODBC) or any other third-party interface.

Note: While CA Workload Automation AE uses the database solely as a SQL engine, it does use Sybase Open Client C Library communications protocol, Oracle SQL*Net V2, or Microsoft SQL Server Multi-Protocol Net-Library to communicate with the vendor database server installation.

Dual Event Servers

You can configure a CA Workload Automation AE instance to run using two event servers (databases), and this configuration is named *dual event server mode*. The dual event server mode provides high availability by running two event servers that are synchronized to maintain identical data, including object definitions and events. CA Workload Automation AE reads from one event server and writes to both the event servers simultaneously. If you lose one event server due to hardware, software, or network problems, operations can continue on the second event server without losing data or functionality. This feature is independent of any replication or redundancy offered by the database.

For various reasons, database users often run multiple instances of servers that are unaware of the other servers on the network. When implementing CA Workload Automation AE, the database can run for CA Workload Automation AE only, or it can be shared with other applications.

Note: For more information about how to install and configure dual event servers, see the *UNIX Implementation Guide* or *Windows Implementation Guide*.

Application Server

The *application server* acts as the communication interface between the event server and the client utilities. It receives requests from the client utilities, queries the event server, and returns the responses to the client utilities.

Scheduler

The *scheduler* is the program, running either as a UNIX daemon process or a Windows service, that runs CA Workload Automation AE. It processes all the events it reads from the event server.

When you start the scheduler, it continually scans the database for events to process. For example, when the scheduler finds a STARTJOB event, it verifies whether the event satisfies the starting conditions for that job in the database. Based on this information, the scheduler determines the actions to take and instructs the appropriate agent to perform the actions. These actions may include starting or stopping jobs, checking for resources, monitoring existing jobs, or initiating corrective procedures.

High Availability

To detect and recover from failure, you can configure CA Workload Automation AE with a second scheduler, named the shadow scheduler. This shadow scheduler must run on a separate computer, and it takes over if the primary scheduler fails. This configuration is named high availability.

If CA Workload Automation AE is running in high availability and dual event server mode, a third scheduler named the tie-breaker scheduler is required. The tie-breaker scheduler is a scheduler process that runs on a third computer. It remains permanently idle and periodically updates its heartbeat in the event servers to indicate its presence. The tie-breaker scheduler resolves contentions and eliminates situations where one scheduler takes over because of network problems.

Note: Shadow and tie-breaker schedulers and dual event servers are independent features. If you configure CA Workload Automation AE to run in high availability mode, these components run together. For more information about shadow and tie-breaker schedulers, installing dual event servers, and configuring high availability, see the UNIX Implementation Guide or Windows Implementation Guide.

Agents and Agent Plug-ins

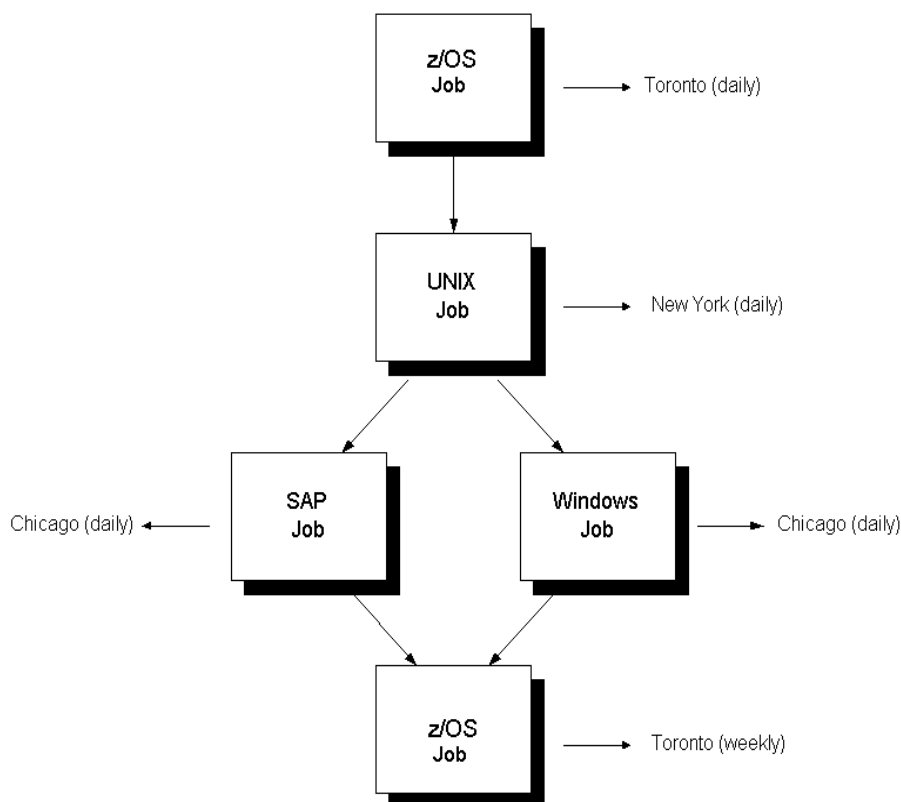
Agents are the key integration components of CA workload automation products. Agents let you automate, monitor, and manage workload on all major platforms, applications, and databases. To run workload on a particular system, you install an agent on that system. If your workload must run on a UNIX computer, for example, you can install and configure the CA WA Agent for UNIX. The agent lets you run UNIX scripts, execute UNIX commands, transfer files using FTP, monitor file activity on the agent computer, and perform many other tasks.

You can extend the functionality of the agent by installing one or more agent plug-ins in the agent installation directory. If you have a relational database such as Oracle, for example, you can install a database agent plug-in to query and monitor the database. Other agent plug-ins are also available. For more information, see the *Implementation Guide* for the appropriate agent plug-in.

Note: The agent plug-ins are only available for UNIX, Linux, and Windows operating environments.

Example: Workload with Different Types of Jobs

The following workload contains z/OS jobs, a UNIX job, an SAP job, and a Windows job, running on different computers, in different locations, and at different times:



Note: For information about agents and agent-plug-ins supported by CA Workload Automation AE, see the *Release Notes*. For information about how to configure CA Workload Automation AE to work with agents, see the *CA Workload Automation AE UNIX Implementation Guide* or *CA Workload Automation AE Windows Implementation Guide*.

Legacy Agent Replaced by CA Workload Automation Agent

The new CA Workload Automation Agent for UNIX, Linux, or Windows replaces the Remote Agent (auto_remote) that was provided with Unicenter AutoSys JM r4.5 and r11. The r11.3 documentation refers to auto_remote as the *legacy agent*.

The new agent provides additional job types, including monitoring and FTP jobs. The agent is automatically installed on the computer where CA Workload Automation AE is installed. You can also install the agent on remote computers to run jobs on those computers.

Note: For information about the differences between the legacy agent and the new agent, see the *Release Notes*.

Client

A *client* is any executable that interfaces with the application server. This includes CA Workload Automation AE Command Line Interface (CLI) applications such as JIL and autorep. It also includes the CA WCC services, which are clients of the application server and service the CA WCC GUI components, and any user-defined binaries that link to the CA Workload Automation AE SDK.

CA Workload Automation AE SDK

The CA Workload Automation AE SDK is a client application, written in Java, C++, or native code, that can be run from any CA Workload Automation AE client installation. Client applications work by calling Application Programming Interfaces (APIs) that are available in the application server. A client can run anywhere in the enterprise provided it can reach the computer where the application server is running. It does not require the installation of a database vendor client. Clients are the means by which users control the scheduling environment by creating and monitoring the scheduling resources.

Note: For more information about the CA Workload Automation AE SDK APIs, see the *API Reference Guide*.

Interface Components

CA Workload Automation AE provides client utilities to help you define, run, and maintain CA Workload Automation AE instances and jobs. The included utilities are platform-specific; however, all platforms include the Job Information Language (JIL) which is part of the Command Line Interface (CLI). The web-based CA WCC interface and the CLI enable you to define, manage, monitor, and report on jobs.

On Windows, CA Workload Automation AE also provides CA Workload Automation AE Administrator using which you can view or modify the configuration parameters of all the CA Workload Automation AE instances that you have installed. You can also define the job profiles that contain the environment variables that must be set for a job to run.

Notes:

- For more information about how to view or modify the configuration parameters of a CA Workload Automation AE instance on Windows using CA Workload Automation AE Administrator, see the *Online Help*.
- For more information about CA WCC, see the CA WCC documentation.
- For more information about JIL, see the *User Guide*. For information about the JIL commands and subcommands, see the *Reference Guide*.

Centralized Control

CA WCC is a web-based user interface that lets you graphically manage, schedule, and monitor CA Workload Automation AE jobs. CA WCC lets you monitor and control workload on different instances of CA Workload Automation AE, providing all the information you need about your jobs from one centralized location. You can create and manage jobs, box jobs, and job flows using CA WCC without needing extensive knowledge of JIL. From within CA WCC, you can also enter job-related and other common JIL commands so that, for example, you can quickly enter large quantities of similar jobs or run autorep without having to move back and forth between the GUI and the CLI.

CA WCC provides your enterprise with multiple applications that let you access status information at a summary and detail level. From the summary information, users can open the appropriate CA WCC applications in context to get detailed information with which to analyze warnings and failure points and take appropriate corrective action. There are additional applications that enable administrators and other users to perform certain system configuration tasks, including publishing core server configurations to other CA WCC servers, running commands on back-end servers, and changing credentials.

With CA WCC, your enterprise has all the information required to efficiently and effectively perform the workload automation function.

Chapter 3: Jobs

This section contains the following topics:

[Scheduling Flexibility](#) (see page 19)

[Job Types](#) (see page 20)

Scheduling Flexibility

All activity controlled by CA Workload Automation AE is based on jobs. A *job* is any single command or executable, UNIX shell script, or Windows batch file. Other objects, such as monitors, reports, and Job Status Console, track job progress. A job is the foundation for the entire operations cycle.

You define jobs to CA Workload Automation AE by creating job definitions. Job definitions control all aspects of job processing, from submission through execution and completion. Each job definition contains attributes that specify the job's properties and behavior. For example, you can specify conditions that determine when and where a job runs. You can base job-starting conditions on a calendar event (basic time/day or custom calendar), job dependency status, or file arrival. CA Workload Automation AE also automatically accounts for international time zone changes. For example, when dependent jobs are running in Tokyo, London, and New York, CA Workload Automation AE recognizes and adapts to each time zone difference, without requiring manual intervention.

CA Workload Automation AE supports numerous job types using agent technology. Agents let you automate, monitor, and manage workload on all major platforms, applications, and databases. To run workload on a particular system, your administrator needs to install an agent on that system. For example, if your workload must run on a UNIX computer, your administrator must install and configure the CA WA Agent for UNIX, which lets you create jobs to run UNIX scripts, execute UNIX commands, transfer files using FTP, monitor file activity on the agent computer, and perform many other tasks.

The CA WA Agent for UNIX, Linux, or Windows is provided with your scheduling manager, giving you a broad set of default job types. The CA WA Agent for z/OS and CA WA Agent for i5/OS are available separately. Agent plug-ins are also available for interfacing with databases, ERP systems, application servers, and web servers.

Note: For more information about the CA WA Agent for UNIX, Linux, or Windows and how to configure it to work with CA Workload Automation AE, see the *CA Workload Automation Agent for UNIX, Linux, or Windows Implementation Guide*.

Job Types

The agents and agent plug-ins let you define and run the following job types:

Agent Name	Supported Job Types
CA Workload Automation Agent for UNIX or Linux	Command (CMD) CPU Monitoring (OMCPU) Disk Monitoring (OMD) File Trigger (FT) File Transfer Protocol (FTP) IP Monitoring (OMIP) Process Monitoring (OMP) Secure Copy (SCP) Text File Reading and Monitoring (OMTF)
CA Workload Automation Agent for Windows	Command (CMD) CPU Monitoring (OMCPU) Disk Monitoring (OMD) File Trigger (FT) File Transfer Protocol (FTP) IP Monitoring (OMIP) Process Monitoring (OMP) Secure Copy (SCP) Text File Reading and Monitoring (OMTF) Windows Event Log Monitoring Jobs (OMEL) Windows Service Monitoring (OMS)
CA Workload Automation Agent for Databases	Database Monitor (DBMON) Database Stored Procedure (DBPROC) Database Trigger (DBTRIG) Structured Query Language (SQL)
CA Workload Automation Agent for i5/OS	i5/OS (I5) All UNIX-based job types if they run in the PASE environment (see the job types listed for CA Workload Automation Agent for UNIX or Linux)
CA Workload Automation Agent for Oracle E-Business Suite	Oracle E-Business Suite Copy Single Request (OACOPY) Oracle E-Business Suite Request Set (OASET) Oracle E-Business Suite Single Request (OASG)

Agent Name	Supported Job Types
CA Workload Automation Agent for PeopleSoft	PeopleSoft (PS)
CA Workload Automation Agent for SAP	SAP Batch Input Session (SAPBDC) SAP BW InfoPackage (SAPBWIP) SAP BW Process Chain (SAPBWPC) SAP Data Archiving (SAPDA) SAP Event Monitor (SAPEVT) SAP Job Copy (SAPJC) SAP Process Monitor (SAPPM) SAP R/3 (SAP)
CA Workload Automation Agent for Application Services	Entity Bean (ENTYBEAN) Hypertext Transfer Protocol (HTTP) Java Remote Method Invocation (JAVARMI) JMS Publish (JMSPUB) JMS Subscribe (JMSSUB) JMX-MBean Attribute Get (JMXMAG) JMX-MBean Attribute Set (JMXMAS) JMX-MBean Create Instance (JMXMC) JMX-MBean Operation (JMXMOP) JMX-MBean Remove Instance (JMXMREM) JMX-MBean Subscribe (JMXSUB) Plain Old Java Object (POJO) Session Bean (SESSBEAN)
CA Workload Automation Agent for Web Services	Plain Old Java Object (POJO) Web Service (WBSVC)
CA Workload Automation Agent for z/OS	z/OS Data Set Trigger (ZOSDST) z/OS Manual (ZOSM) z/OS Regular (ZOS)

Notes:

- The Command, Box, File Watcher, and User-defined job types are still supported in CA Workload Automation AE r11.3.
- For more information about how these job types work, see the *User Guide*. For detailed information about the JIL syntax used to define these jobs, see the *Reference Guide*.
- You can also use CA WCC to define jobs. For more information about using CA WCC to define the job, see the CA WCC documentation.

Chapter 4: Security

This section contains the following topics:

[Overview](#) (see page 23)

[System-Level Security](#) (see page 23)

[Native Security](#) (see page 25)

[External Security](#) (see page 26)

[Modifying Your Security After Installation](#) (see page 26)

Overview

CA Workload Automation AE includes features that let you secure objects such as jobs, calendars, cycles, global variables, machines, and resources. You can delegate administrative privileges to these objects to specific users or user groups.

CA Workload Automation AE provides security in the following ways:

- System-level security
- Native security
- External security

Note: We recommend that you have a thorough understanding of CA Workload Automation AE security as it pertains to CA EEM. For more information about controlling the security settings with the native `autosys_secure` command, see the *Reference Guide*. You can use this chapter with the commands in the *Reference Guide* and the security information in the *CA Workload Automation Security Guide*.

System-Level Security

System-level security prevents the following:

- Unauthorized access to job information.
- Unauthorized jobs from running on a machine.

CA Workload Automation AE provides the following security features at the system level:

- Database field verification
- Job definition encryption
- Remote authentication
- User and database administrator passwords
- File system access restriction
- Data encryption

Note: Job definition encryption applies to legacy agents only.

These security features are always available and are in effect regardless of the active security mode.

Note: For more information about CA Workload Automation AE system-level security, see the *CA Workload Automation Security Guide*.

Encryption and FIPS 140-2 Compliance

CA Workload Automation AE uses Advanced Encryption Standard (AES) encryption to comply with the U.S. Government encryption standard FIPS 140-2. This standard requires a FIPS-certified library and FIPS-certified cipher algorithm, such as AES.

CA Workload Automation AE uses the AES cipher algorithm to encrypt and decrypt data shared between the command line utilities, agent, scheduler, and the application server. Encryption requires eTrust Public Key Infrastructure (ETPKI), which is automatically installed with the server, agent, or client.

AES also requires an encryption key. You can modify the encryption key by using the `as_config` command for the following components:

- Application server and client utilities—The key is stored in the `$AUTOUSER/cryptkey.txt` file.
- Application server and agent—The key is specified in the machine definition for the agent. This key must match the key in the agent's `cryptkey.txt` file. The `cryptkey.txt` file is located in the `install_directory/SystemAgent/agent_name` directory.
- Scheduler and agent—The key is specified in the machine definition for the agent. This key must match the key in the agent's `cryptkey.txt` file. The `cryptkey.txt` file is located in the `install_directory/SystemAgent/agent_name` directory. For the agent on z/OS, the encryption key is stored in the `cryptkey_alias.txt` file located in the `$AUTOUSER.instance_name` directory (the default).

Note: For more information about data encryption, see the *CA Workload Automation Security Guide*. For more information about the `cryptkey_alias.txt` file, see the *UNIX Implementation Guide* or *Windows Implementation Guide*.

Native Security

Native security is the default security mode that CA Workload Automation AE runs under. In native security mode, you can do the following to secure objects and delegate administrative privileges to users:

- Define EDIT superusers who have administrative privileges.
- Define EXEC superusers who have the authority to issue the `sendevent` command to send the execute events that affect the running of a job or the state of a job.
- Manage user IDs that jobs run under (job owners).
- Modify edit and execute permissions on a job-by-job basis.

Important! Although native security mode provides a level of security for certain objects and activities, the level of protection that native security mode provides is limited compared to that of external security. Only external security lets you control role-based access to objects (such as jobs, calendars, cycles, global variables, machines, and resources) at a granular level. Instead of using native security, we recommend that you enable external security after installation by configuring CA Workload Automation AE to work with CA EEM.

Note: For more information about native security or how to configure CA Workload Automation AE to work with CA EEM, see the *CA Workload Automation Security Guide*.

External Security

External security is enabled by integrating CA Workload Automation AE with CA EEM. The external security mode is robust and provides better flexibility than the native security mode.

An EDIT superuser can enable external security by using the `autosys_secure` command. When external security mode is enabled, CA EEM is used to assign administrative rights to a user to define policies and to check whether a given user can switch the security mode of CA Workload Automation AE back to native. CA EEM lets you manage your user base, create roles for your enterprise, and assign roles to users. It also maintains security policies that govern what objects can be accessed by which users.

Notes:

- All GUI applications and all command line interfaces have callouts to security. User-defined resource classes in CA EEM are used to govern what types of resources can be controlled by which users. For more information about these resource classes, see the *CA Workload Automation Security Guide*.
- While external security mode is enabled, native security is not enforced. For more information about external security or native security, see the *CA Workload Automation Security Guide*.

Modifying Your Security After Installation

Native security is enabled by default. After CA Workload Automation AE is installed, you can use the `autosys_secure` command to enable external security. Option [1] of `autosys_secure` lets you activate or deactivate CA EEM for external security. Option [2] lets you set the EDIT and EXEC superuser privileges for native security.

Notes:

- If appropriate, both privileges can be assigned to a single user. The EDIT and EXEC superusers must be valid users on the host or domain that you are using.
- For more information about the full capabilities of the `autosys_secure` command, see the *Reference Guide*.
- For more information about EDIT or EXEC superusers or how to enable external security after CA Workload Automation AE is installed, see the *CA Workload Automation Security Guide*.

Chapter 5: Key Features

This section contains the following topics:

[Fault-Tolerant Operation and Workload Balancing](#) (see page 27)

[Cross-Instance and Cross-Platform Scheduling](#) (see page 29)

[Cross-Instance Job Dependencies with CA Workload Automation EE](#) (see page 31)

[Real and Virtual Resources](#) (see page 31)

[Integration with Other CA Products](#) (see page 32)

[Monitoring and Reporting Jobs](#) (see page 32)

Fault-Tolerant Operation and Workload Balancing

With CA Workload Automation AE, you are assured automated, uninterrupted scheduling services—a key requirement for business and other mission-critical operations—through CA Workload Automation AE high availability and dual event servers. This feature lets you run the scheduler and the database on two computers as a backup system. If one computer goes down, the other detects the problem, and performs an automatic rollover, picking up where the first computer left off.

Before CA Workload Automation AE starts each job, it automatically checks computer and resource availability, issuing an alarm and scheduling a restart if a problem is detected. It continuously monitors each job as it runs, and verifies its completion.

CA Workload Automation AE uses the construct of virtual computers to group real computers for workload balancing. Thus, the best computer for the job is automatically selected at run time, giving you an effective way to manage resources across your network.

High Availability: Shadow Scheduler

CA Workload Automation AE lets you set up a second scheduler, named the *shadow scheduler*. Each scheduler must run on a separate computer.

Both the primary scheduler and the shadow scheduler periodically update the event server to indicate that they are in active mode. The shadow scheduler typically stays in idle mode, checking the event server for routine database updates from the primary and tie-breaker schedulers, which indicates that workload scheduling is processing normally. If the shadow scheduler stops seeing updates to the event server, it assumes that the primary scheduler has failed.

When the shadow scheduler does not see an update from the primary scheduler, it checks for the tie-breaker scheduler update to the event server. If it cannot find an update, the shadow scheduler shuts down. If it can, the shadow scheduler tries to signal the primary scheduler to stop and takes over event processing.

Note: For more information about configuring high availability, see the *UNIX Implementation Guide* or *Windows Implementation Guide*.

High Availability and Dual Event Servers: Tie-breaker Scheduler

When you run high availability using dual event servers, a third scheduler named *tie-breaker scheduler* is required. Without the tie-breaker scheduler, both the primary and shadow schedulers assume that the other scheduler has failed and, therefore, both proceed with processing events.

For example, imagine a scenario where the shadow scheduler is configured to run on the same computer as one of the event servers, and this computer gets disconnected from the network. The shadow scheduler continues to use the event server on its node assuming there has been an event server failure and that the primary scheduler has failed. However, it is actually the shadow scheduler computer that has failed.

The tie-breaker scheduler running on a third node resolves this problem. It remains permanently idle and updates the event servers periodically to indicate its presence. In the example scenario, the shadow scheduler realizes that it is the failed node when it does not receive updates from the tie-breaker scheduler.

Cross-Instance and Cross-Platform Scheduling

CA Workload Automation AE supports the following:

- Cross-Instance scheduling
- Cross-Platform scheduling

Note: For more information about configuring cross-instance or cross-platform scheduling, see the *UNIX Implementation Guide* or *Windows Implementation Guide*.

CA Workload Automation AE Cross-Instance Job Dependencies

A CA Workload Automation AE *instance* is one licensed version of CA Workload Automation AE software running as a server and as one or more clients, on one or more computers. An instance uses its own scheduler, one or more application servers, and event server, and operates independently of other instances.

Different instances can run from the same executables and can have the same value for \$AUTOSYS. However, each instance must have different values for \$AUTOUSER and \$AUTOSERV. Different instances can also be run on the same computer.

Multiple CA Workload Automation AE instances are not connected, but they can communicate with one another. This communication lets you schedule workload across instances in your enterprise. You can define jobs that have dependencies on jobs running on other instances (*cross-instance job dependencies*). A CA Workload Automation AE job with these dependencies conditionally starts based on the status of the job on the other instance. In this situation, your instance's scheduler acts as a client and issues sendevent commands to the external instance. The other instance's application server processes the sendevent request and stores the dependency request or status update in its database.

You can also manually send events from one instance to another.

Cross-Platform Scheduling

Cross-platform scheduling lets you schedule and reroute jobs between CA Workload Automation AE and other machines running on different platforms, including mainframe.

To use cross-platform scheduling, required components must be installed on the CA Workload Automation AE computer and on the external machine that CA Workload Automation AE works with. The scheduling manager or remote machine must also be defined as an *external instance* in the CA Workload Automation AE database.

Bi-Directional Scheduling

CA Workload Automation AE supports *bi-directional scheduling*, which lets you start jobs from remote machines (inbound) or submit jobs on remote machines (outbound).

With *inbound job scheduling*, CA Workload Automation AE acts as an agent and accepts job submissions from remote machines or other scheduling managers (such as CA Jobtrac Job Management and CA Workload Automation SE). The jobs are defined and run on the CA Workload Automation AE instance that is acting as an agent.

With *outbound job scheduling*, CA Workload Automation AE acts as a scheduling manager and sends job submissions to remote machines. The jobs are defined on the CA Workload Automation AE instance that is acting as a scheduling manager. The jobs run on the remote machine or other scheduling manager.

For example, a Linux Oracle instance can initiate jobs in a Windows Microsoft SQL Server instance, or a Windows Microsoft SQL Server instance can initiate jobs in a Solaris Oracle instance. You can add additional instances, such as Solaris Sybase, AIX Oracle, or HP Oracle instance, to the environment.

The CA Workload Automation AE cross-platform interface controls the bi-directional scheduling mode. You can configure the cross-platform interface to enable the following modes:

- Outbound job scheduling
- Inbound and outbound job scheduling (bi-directional scheduling)
- No cross-platform scheduling (the default)

Note: There are no restrictions on platforms, event servers, or number of instances when running in bi-directional scheduling mode.

CA UJMA and CA AutoSys WA Connect Option Dependencies

You can define jobs that have dependencies on jobs running on external machines (*external job dependencies*). These machines must be defined as *external instances* on CA Workload Automation AE. A CA Workload Automation AE job with these dependencies conditionally starts based on the status of the job running on the other instance. These dependencies are jobs that execute on an external instance but were not initiated on behalf of the local CA Workload Automation AE instance. When the dependent job completes, status information is sent to the local CA Workload Automation AE instance and recorded in the database. For example, a job in one instance can be defined to start based on the status of jobs running on a mainframe system.

Cross-Instance Job Dependencies with CA Workload Automation EE

You can define and monitor cross-instance (external) job dependencies between CA Workload Automation AE and CA Workload Automation EE. These job dependencies let you create job flows between distributed and mainframe systems.

For more information about how external job dependencies work, see the *User Guide*. For detailed information about the commands and JIL syntax used to define external job dependencies, see the *Reference Guide*.

Real and Virtual Resources

You can define resources to CA Workload Automation AE and specify those resources as job dependencies. Using resources can help improve your environment's performance. For example, you can prevent jobs from running simultaneously and help ensure that a job is submitted only when the minimum number of resources is available.

You can define two types of resources:

Real resources

Defines system conditions that are directly tied to a physical system (for example, physical memory).

Virtual resources (depletable, renewable, and threshold)

Defines representations that cannot be physically measured and are not directly tied to a physical system. You can define a virtual resource by specifying a name, machine, and number of units.

Note: For more information about how resources and resource job dependencies work, see the *User Guide*. For detailed information about the JIL syntax used to define resources and resource job dependencies, see the *Reference Guide*.

Integration with Other CA Products

CA Workload Automation AE integrates with other CA products to perform the following tasks:

- CA Embedded Entitlements Manager (CA EEM)—Lets you secure objects such as jobs, calendars, cycles, global variables, machines, and resources, and delegate role-based access to the users in your enterprise.
- Secure Socket Adapter (SSA)—Lets CA components use a single multiplexed communication port to ease firewall administration and minimize conflicts with other applications.
- (Windows only) Notification Services component of CA NSM—Lets you send wired and wireless messages, using protocols and devices, to operators or administrators who resolve problems or attend to emergencies.
- CA NSM Event Management—Routes workload automation messages to a consolidated event console. You can automate manual problem resolution tasks, filter and consolidate multiple events, monitor for unusual conditions, and take proper corrective action.
- CA Service Desk—Lets you open a service desk ticket (request or incident) when a job fails. From the CA Service Desk web interface, you can submit tickets, check status, and browse the knowledge base.
- CAICCI—Enables communication with mainframe products, legacy agents for mainframe systems, and Event Management.
- CA Spectrum Automation Manager—Lets you monitor and provision machines.

Note: For information about how to configure CA Workload Automation AE to work with other CA products, see the *CA Workload Automation AE UNIX Implementation Guide* or *CA Workload Automation AE Windows Implementation Guide*.

Monitoring and Reporting Jobs

Monitors provide a real-time view of the system. *Reports* (sometimes called browsers) let you use a variety of reporting views to examine historical information about job executions.

Monitors and reports are simply applications that run against the database. Because all information is in the database, monitors and reports that retrieve information from the database provide a complete picture of the state of the entire system. Monitors and reports can run with any database, including dual event servers, and on any CA Workload Automation AE client computer.

Monitors and reports let you filter and display only the information you are interested in from a vast collection of data. That is, they are tools that can provide information meaningful to you.

Both monitors and reports filter events by type and by job or groups of jobs. Reports also filter by time (monitors do not filter by time because they provide real-time information).

Notes:

- Because monitors provide a picture of the system's state in real time, they will not provide any information if the scheduler is down. On the other hand, reports provide a picture of the system's state from a historical perspective, not in real time.
- For more information about monitors and reports, see the *User Guide*.

Index

A

agent plug-ins • 15

C

CA WCC • 17

CA Workload Automation AE

architecture • 12

components • 12

cross-platform scheduling • 29

distinctive features • 9

high availability options • 28

integrating with other products • 32

job dependencies • 30, 31

job types • 20

monitoring and reporting jobs • 32

overview • 7

resources • 31

scheduling jobs • 19

security • 23

workload balancing • 27

CA Workload Automation AE SDK • 16

CA Workload Automation AE security

encryption • 24

external security • 26

FIPS compliance • 24

modifying after installation • 26

native security • 25

overview • 23

system-level • 23

clients • 16

components

agents • 15

application servers • 14

event servers • 13

schedulers • 14

cross-platform scheduling • 29

D

dual event servers • 13

E

encryption • 24

external security • 26

F

FIPS compliance • 24

H

high availability options • 14, 28

I

instances • 11

integrating with other products • 32

interface components • 17

J

job dependencies • 30, 31

job types • 20

L

legacy agents • 16

M

monitoring and reporting jobs • 32

N

native security • 25

R

real resources • 31

S

scheduling jobs • 19

system-level security • 23

V

virtual resources • 31

W

workload automation • 8

workload balancing • 27