CONTROL-M/Agent

for UNIX and Microsoft Windows

Administrator Guide

Supporting

CONTROL-M/Agent for UNIX and Microsoft Windows 6.3.01

December 2006

Contacting BMC Software

You can access the BMC Software website at http://www.bmc.com. From this website, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

United States and Canada

Address	BMC SOFTWARE INC 2101 CITYWEST BLVD HOUSTON TX 77042-2827 USA	**Telephone**	713 918 8800 or 800 841 2031	**Fax**	713 918 8000

Outside United States and Canada

Telephone	(01) 713 918 8800	**Fax**	(01) 713 918 8000

Restricted rights legend

Customer support

You can obtain technical support by using the BMC Software Customer Support website or by contacting Customer Support by telephone or e-mail. To expedite your inquiry, see “Before contacting BMC.”

Support website

You can obtain technical support from BMC 24 hours a day, 7 days a week at http://www.bmc.com/support_home. From this website, you can

- read overviews about support services and programs that BMC offers
- find the most current information about BMC products
- search a database for issues similar to yours and possible solutions
- order or download product documentation
- report an issue or ask a question
- subscribe to receive proactive e-mail alerts when new product notices are released
- find worldwide BMC support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support by telephone or e-mail

In the United States and Canada, if you need technical support and do not have access to the web, call 800 537 1813 or send an e-mail message to support@bmc.com. Outside the United States and Canada, contact your local support center for assistance.

Before contacting BMC

Have the following information available so that Customer Support can begin working on your issue immediately:

- product information
  - product name
  - product version (release number)
  - license number and password (trial or permanent)
- operating system and environment information
  - machine type
  - operating system type, version, and service pack or other maintenance level such as PUT or PTF
  - system hardware configuration
  - serial numbers
  - related software (database, application, and communication) including type, version, and service pack or maintenance level
- sequence of events leading to the issue
- commands and options that you used
- messages received (and the time and date that you received them)
  - product error messages
  - messages from the operating system, such as `file system full`
  - messages from related software

Contents

Figures

Tables

About this book

This book describes CONTROL-M/Agent administration and provides information about parameters and utilities.

Related publications

The following related publications supplement this book:

Category	Document	Description
installation document	*CONTROL-M/Server and CONTROL-M/Agent Installation Guide*	describes the installation of CONTROL-M components
administration	*CONTROL-M/Server for UNIX and Microsoft Windows Administrator Guide*	describes setup, maintenance, security, and utilities for CONTROL-M/Server on a specific type of computer
	CONTROL-M/Agent for UNIX and Microsoft Windows Administrator Guide	describes configuration and maintenance of CONTROL-M/Agent on a specific type of computer.
	CONTROL-M/Enterprise Manager Administrator Guide	describes tasks that the CONTROL-M/EM administrator must perform to define, monitor, and maintain the CONTROL-M/EM environment
	CONTROL-M/eTrigger Administrator Guide	describes how to trigger job submission and tracking activities using a web-based (HTML) interface
user information	*CONTROL-M Job Parameter and Variable Reference Guide*	describes syntax and usage for all parameters and variables that are included in CONTROL-M/Server job processing definitions
	CONTROL-M/Enterprise Manager Utility Guide	describes command-line utilities that can be used to perform various CONTROL-M/EM tasks in batch mode
	CONTROL-M/Enterprise Manager User Guide	describes CONTROL-M/EM concepts, features, facilities, and operating instructions

Category	Document	Description
customization	*CONTROL-M Language Customization Guide*	configuration procedures for running CONTROL-M/EM, CONTROL-M/Desktop, CONTROL-M/Server, CONTROL-M/Agent, and BMC Batch Impact Manager with different languages
	SSL for CONTROL-M Administrator Guide	describes the use of the Secure Sockets Layer (SSL) protocol to authenticate and encrypt communications in CONTROL-M

Conventions

Text and examples are given according to UNIX usage, unless otherwise stated. The default home directory of the UNIX user account under which CONTROL-M/Agent is installed is <*agentHome*>.

The following abbreviations are used in this guide:

Abbreviation	Description
CM	CONTROL-M/Control Module A product library for a specific application or operating system used by CONTROL-M/Agent to support that application or operating system.
CONTROL-M/EM	CONTROL-M/Enterprise Manager

The following table explains conventions for syntax statements and provides examples:

Item	Example
Items in italic type represent variables that you must replace with a name or value. If a variable is represented by two or more words, initial capitals distinguish the second and subsequent words.	*alias* *databaseDirectory* *serverHostName*
Brackets indicate a group of optional items. Do not type the brackets when you enter the option. A comma means that you can choose one or more of the listed options. You must use a comma to separate the options if you choose more than one option.	[*tableName*, *columnName*, *field*] [-full, -incremental, -level] (Unix)

Item	Example
Braces indicate that at least one of the enclosed items is required. Do not type the braces when you enter the item.	{*DBDName* | *tableName*} UNLOAD device={disk | tape, *fileName* | *deviceName*} {-a | -c} (Unix)
A vertical bar means that you can choose only one of the listed items. In the example, you would choose either *commit* or *cancel*.	{commit | cancel} {-commit | -cancel} (Unix)
An ellipsis indicates that you can repeat the previous item or items as many times as necessary.	*columnName* . . .

Chapter 1

Understanding CONTROL-M/Agent

CONTROL-M / Agent describes the concepts and the tools that you will use to set up and manage CONTROL-M / Agent on UNIX and Microsoft Windows computers. The CONTROL-M / Agent for UNIX and Windows product is a component of the CONTROL-M scheduling solution (Figure 1 and Table 1 on page 16.)

CONTROL-M / Server manages production control and scheduling, and submits and tracks jobs across your network.

Starting with version 6.3.01, CONTROL-M / Agent can run jobs on remote hosts. For more information, see the *CONTROL-M/Server for UNIX and Microsoft Windows Administrator Guide* version 6.3.01 and *CONTROL-M/Enterprise Manager Administrator Guide* version 6.3.01 Fix Pack 1.

CONTROL-M / Agent submits jobs for execution on the agent computer, monitors the jobs, and analyzes the output of files. The completion status of jobs and the results of post-processing analysis are transmitted back to CONTROL-M / Server.

Information about installing and upgrading CONTROL-M / Agent is described in the *CONTROL-M/Server and CONTROL-M/Agent Installation Guide*.

Figure 1 CONTROL-M scheduling solution

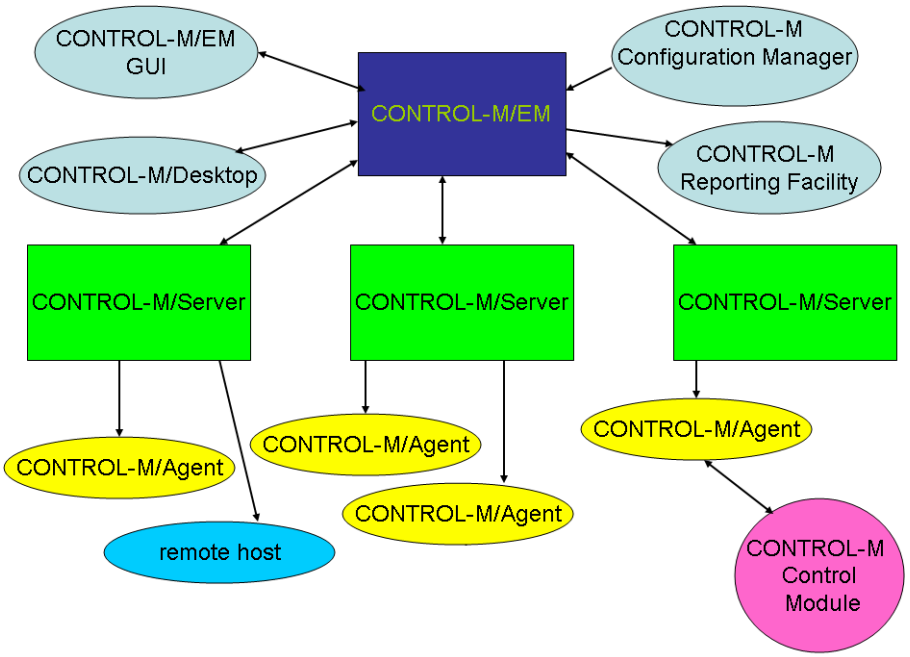


Table 1 CONTROL-M products

Product	Description
CONTROL-M / EM	GUI-based control center from which you can manage all scheduling and workflow activities
CONTROL-M / Server	engine (on a UNIX or Microsoft Windows computer) that drives scheduling in a data center Each CONTROL-M / Server can manage multiple CONTROL-M / Agents on various computers.
CONTROL-M / Agent	software responsible for job submission and execution.
CONTROL-M / eTrigger	web-based product that creates and runs jobs under CONTROL-M using input from a web page.
CONTROL-M / Control Module	control modules that enable CONTROL-M / Agents to interface with other applications (for example, SAP and Oracle Applications)

CONTROL-M/Agent functions

CONTROL-M / Agent manages the following types of job handling requests:

- instructions to submit a job on the agent computer and remote hosts
- requests for information about jobs that have been submitted, are currently executing, or have recently completed on the agent computer

- requests to view or edit job script statements
- requests to view job output (SYSOUT) or job documentation
- requests to kill jobs that are currently executing

In addition, CONTROL-M / Agent can manage job output (SYSOUT) and issue Shout messages according to job processing parameters that are supplied with a job submission request.

CONTROL-M/Agent concepts

Table 2 describes key concepts related to CONTROL-M / Agent.

Table 2 CONTROL-M/Agent key concepts

Item	Description
configuration parameters	CONTROL-M / Agent configuration parameters are used to assign values during the installation procedure. BMC Software recommends that you use the Agent Configuration utility (see page 53) to modify these parameters after installation. ■ (*UNIX*) Table 23 on page 83 lists the parameters that are in the **CONFIG.dat** file and their default values. ■ (*Windows*) Table 17 on page 75 lists the parameters and their default values.
command_line utilities	Many CONTROL-M tasks by using command_line utilities. For more information, see Chapter 5, “Utilities.”
Control Modules	A Control Module (CM) is a DLL component (*Windows*) or shared object (*UNIX*) that enables CONTROL-M / Agent to communicate with other applications in your data center. For example, the CM for SAP communicates between CONTROL-M / Agent and SAP. ■ Installing CONTROL-M / Agent installs a CM for UNIX and Windows operating systems and assigns values to the configuration parameters. ■ BMC Software recommends that you use the Agent Configuration utility (see page 53) to modify these parameters after installation. ■ (*UNIX*) Table 24 on page 86 lists the parameters that are in the **OS.dat** file and their default values. ■ (*Windows*) Table 25 on page 88 lists these parameters and their default values.

Server and agent communication

More than one CONTROL-M / Agent can reside on a computer. Thus multiple CONTROL-M / Servers can communicate with different agents on the same computer. For example, a CONTROL-M / Server can submit a job to an agent in a test environment while another CONTROL-M / Server sends an accounts payable job to the same computer, using a different agent.

NOTE

If more than one agent is installed, each agent must have different server_to_agent ports. For more information, see "Agent configuration utility" on page 53.

You can configure CONTROL-M / Agent to work with primary and backup CONTROL-M / Servers. If the primary server fails and defers to a backup server, the agent defers to the same backup server.

Connecting CONTROL-M/Agents on the same host to a CONTROL-M/Server

CONTROL-M / Agent can support multiple agents on the same computer, connected to the same CONTROL-M / Server but using different server_to_agent ports.

To configure your system

1 Ensure that the values in the server_to_agent port field of the agent Configuration utility are different for each agent.

For more information, see "Agent configuration utility" on page 53.

2 Define the second agent on CONTROL-M / Server using a different logical name and port from step 1.

3 On the computer on which the CONTROL-M / Server to which you want to add the agent is installed, log on as root (*UNIX*) or as an administrator (*Windows*)

4 Open the **hosts** file located in the **etc** directory and add the following line

<IP address of the Agent computer> <logical name>

5 On the second agent, change the value of the **Logical Agent name** field in the Advanced window of the Agent Configuration utility.

EXAMPLE

If two agents are installed on a computer called appserver with the IP address 11.22.33.44, complete the following steps:

1. On the server, define the second agent as appserver2.
2. Add the line 11.22.33.44 appserver2 to the **/etc/hosts** file.
3. Change the **Logical Agent name** field for the second agent to **appserver2**.

Communicating with a specific CONTROL-M/Agent (*Windows only*)

Each CONTROL-M / Agent that resides on the same computer has a different Server-to-Agent listening port. CONTROL-M / Server communicates with a specific CONTROL-M / Agent by referring to its listening port.

The -agent <agent name> parameter is used in CONTROL-M / Agent utilities to specify which CONTROL-M / Agent will manage that utility. The *<agent name>* variable represents the name of the CONTROL-M / Agent specified during the installation procedure.

For most utilities if the -agent parameter is not specified, the default CONTROL-M / Agent for that computer is used. However, if a configuration utility is run without specifying the -agent parameter, the user is prompted to select the CONTROL-M / Agent. For more information about invoking CONTROL-M / Agent utilities, see "Invoking a utility on a computer with multiple agents" on page 43.

When running utilities on a Windows cluster, you must always specify the -agent parameter since there is no default agent in the cluster environment.

The default CONTROL-M / Agent is determined by upgrading an earlier version of CONTROL-M / Agent, or during the installation procedure, as described in the Table 3.

Table 3 Determining the default CONTROL-M/Agent

Method	Description
Upgrade	When upgrading CONTROL-M / Agent from version 6.1.03 or 6.2.01, the upgraded agent maintains its status. Meaning, if the upgraded agent was the default agent, it will remain the default agent.
Installation Procedure	During installation of the first CONTROL-M / Agent, the user can select the agent as default. ■ If selected, this CONTROL-M / Agent is set as **Default**. If the user specifies a name for the agent, the agent must be referred to by that name when specifying the -agent parameter. For more information about the -agent parameter, see Chapter 4, "Utilities".
	Note: The name **Default** (not case sensitive) is a reserved name.
	■ If not selected, a subsequent CONTROL-M / Agent can be selected as default.

After the default CONTROL-M / Agent is installed on a computer, the installation procedure prompts for a name (free text, not case sensitive) for each subsequent CONTROL-M / Agent that is installed on the same computer.

Agent to Server connection models

There are two possible models to guide you how you can connect to CONTROL-M / Server.

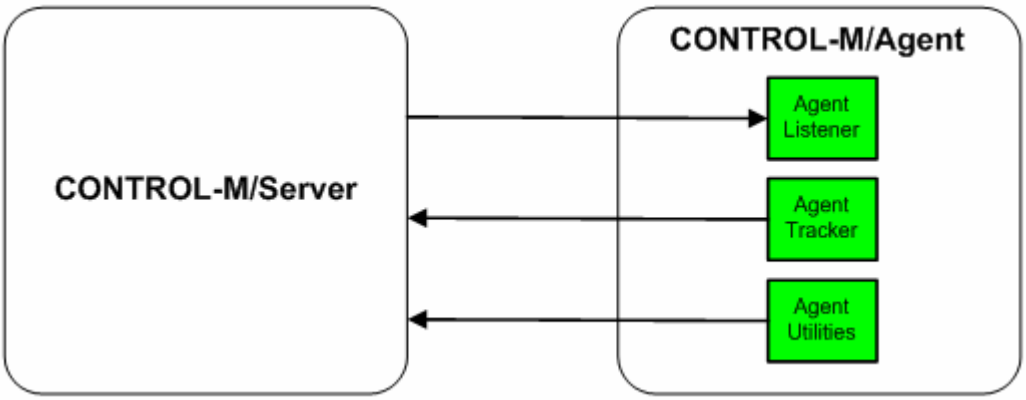
- Transient connection - default model used with new and upgrade installations. For more information, see "Transient connection model" on page 21.
- Persistent connection model - optional model with improved connectivity between the Server and Agent. For more information, see "Persistent connection model" on page 21

For more information about the connection model parameters, see the Persistent Connection and Allow_Comm_Init parameters in Table 11 on page 55.

NOTE

UNIX computers work with processes, while Windows computers use services. In this section, these terms are interchangeable.

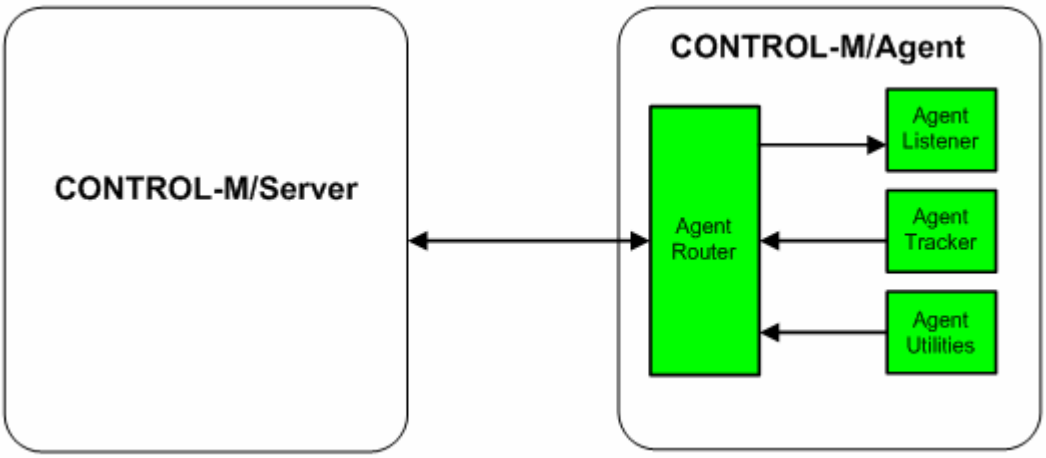
Transient connection model



In the transient connection model, CONTROL-M / Server initiates a connection with the CONTROL-M / Agent Listener process to submit jobs and other action requests. In contrast, the Agent Tracker and Agent Utilities only open a connection to CONTROL-M / Server when they need it. Once the purpose for opening these connections is finished, the connection terminates.

However, if CONTROL-M / Server sits behind a firewall, the Agent Tracker and Agent Utilities are not able to open a connection to the server. As a result, Agent Utilities cannot be run and job statuses are updated only upon server request, approximately once every 15 minutes.

Persistent connection model



In the persistent connection model, the connection between the server and agent is constant and can be initiated by both the server and agent. Upon startup of the agent, the Agent Router process is started and acts as a broker between the other agent components and the server.

The Agent Router process allows CONTROL-M/Server to maintain a constant connection with the agent. However, when CONTROL-M/Server sits behind a firewall, the Agent Router cannot initiate the connection with the server. Once the server creates the connection to the Agent Router, the Agent Tracker and Agent Utilities processes use this connection to communicate freely with the server.

Chapter 2

Implementing CONTROL-M/Agent

This chapter describes procedures and facilities that you can use to keep CONTROL-M/Agent running efficiently. The following topics are discussed in this chapter:

Starting and stopping CONTROL-M/Agent

This section provides information about starting and stopping CONTROL-M/Agent, as well as maintaining the agent services.

For UNIX

You can start the agent automatically (whenever the computer starts) or manually.

To start in CONTROL-M/Agent and Tracker automatically

To start CONTROL-M/Agent automatically after the computer on which CONTROL-M/Agent is installed is restarted, copy the **rc.agent_user** startup script from ctm/scripts in the directory under which you installed CONTROL-M/Agent and place it in the operating system's startup script directory.

To start CONTROL-M/Agent and Tracker manually

1 Log on to the agent computer as **root**.

2 Enter the following command from the agent directory or run the command interactively:

<agent_path>ctm/scripts/start-ag –u <agent_username> -p ALL :

To stop CONTROL-M/Agent and Tracker manually

1 Log on to the agent computer as **root**.

2 Enter the following command from the agent directory or run the command interactively:

<agent_path>ctm/scripts/shut-ag –u <agent_username> -p ALL :

This command shuts down all CONTROL-M / Agent processes.

For Windows

The following CONTROL-M / Agent services are installed during the CONTROL-M / Agent installation procedure:

- CONTROL-M Listener Service
- CONTROL-M Tracker Service
- CONTROL-M Router Service
- CONTROL-M FileWatcher Service

CONTROL-M Listener and Tracker services

The CONTROL-M / Agent Listener and CONTROL-M / Agent Tracker services can run as Microsoft Windows background processes that are transparent to the user. These services remain active as long as Microsoft Windows is running. For more information, see “Starting and stopping Listener and Tracker services” on page 26.

To specify the user account for the Agent Listener and Tracker services

1 In the **Services** window, right-click on the relevant agent service and select **Properties**.

2 The **Log On** tab in the **Properties** window prompts you to select **Local System account** or **This Account**.

- If **Local System account** is selected, the service will run in the administrative group and in the native system account environment. By installation default, the following options are selected:
  - (Log on as:) "Local System Account"
  - "Allow service to interact with desktop"

  These options enable the Listener and Tracker services to open windows in the Microsoft Windows desktop. However, the Local System Account cannot access files across a network and cannot send a Shout message to an e-mail destination.

- If the owner of any jobs run by CONTROL-M/Agent has a "Roaming Profile", select **This Account** mode. If **This Account** is selected, specify an Administrators group user and password. The service will run in the specified user environment. The format for the value in the **This Account** text box is **<Domain>\<User>**. For more information, see "Support for a Roaming Profile"on page 25.

NOTE

You must use **This Account** mode to run jobs on remote hosts using the WMI connection.

NOTE

The administrator selected as part of **This Account**, must have the following permissions in the Local Security Settings window:

- Act as part of the operating system (Windows 2000 users, only)
- Increase quotas
- Replace a process level token
- Log on as a service

The service's log on account must be a member of the **Local Administrative Group**.

On Windows 2000, the **Act as part of operating user** privilege is granted to the account.

Support for a Roaming Profile

CONTROL-M/Agent support for Roaming Profile requires the following:

- The profile must reside on the network. If the network path includes the environment variable, CONTROL-M / Agent expands the path and loads the User Profile from the expanded path.

- After loading the user profile, CONTROL-M / Agent sets all environment variables from the roaming profile:

  — Logs into a different computer with the roaming user
  — Changes / adds the private environment variable
  — Logs out
  — Runs the CONTROL-M job on the original computer

- New environment variables or updated variables performed by the roaming user on any computer will be detected by the CONTROL-M job.

Starting and stopping Listener and Tracker services

The Listener and Tracker services operate as background processes that are not visible to the user. When the computer is turned on or rebooted, these services start automatically if the Startup Type configuration parameter is set to **Automatic**. (See "Startup Type" on page 80.) These services remain active as long as Microsoft Windows is running.

If you stop these services, you can restart them manually (as described below), or you can reboot the system (if Startup Type is **Automatic**). If the Startup Type is set to **Manual**, you must start these services manually (as described below).

You can modify the Startup Type value. BMC Software recommends that the Listener and Tracker services be treated the same way. The recommended Startup Type is **Automatic**.

Starting the Listener and Tracker services

1 Select **Start => Settings => Control Panel => Administrative Tools**.

2 In the Administrative Tools window, click **Services**.

3 Select the **CONTROL-M/Agent Listener** service and click **Start**.

4 Select the **CONTROL-M/Agent Tracker** service and click **Start**.

Stopping CONTROL-M Agent and Tracker Services

1 Select **Start => Settings => Control Panel => Administrative Tools**.

2 In the Administrative Tools window, click **Services**.

3 Select the **CONTROL-M/Agent Listener** service and click **Stop**.

4 Select the **CONTROL-M/Agent Tracker** service and click **Stop**.

CONTROL-M/Agent Router service

The Router service acts as a broker between different agent components and CONTROL-M / Server. This service only runs when CONTROL-M / Agent is configured to work in Persistent mode. For more information, see "Persistent connection model" on page 21.

By default, the Router service **Startup Type** is set to Manual. BMC Software recommends that you not change this setting.

CONTROL-M/Agent FileWatcher service

BMC Software recommends that you do not make any changes to this service.

Determining the owner of jobs run on the agent (*Windows only*)

CONTROL-M / Agent runs a job in the environment of the owner of the job with the permissions granted to that owner.

The owner of a CONTROL-M / Agent job is determined by the value of the **Logon As User** configuration parameter. This parameter can be set either during the installation (see the *CONTROL-M/Server and CONTROL-M/Agent Installation Guide*) or after the installation, as described in the Agent Configuration utility on page 53).

- If **Logon As User** is set to **Y**, the owner of the job is the owner specified in the CONTROL-M job definition.

  To enable a specific user to run (be the owner of) CONTROL-M jobs, the password of the user must be entered into CONTROL-M using the ctmpwd utility. For more information, see "ctmpwd utility" on page 74.

- If **Logon As User** is set to **N,** the owner of the job is the user account for the CONTROL-M / Agent service:

  — **Local System**

  — **This Account**

For more information, see "To specify the user account for the Agent Listener and Tracker services" on page 25.

The following requirements must be satisfied:

- If a user with a "Roaming Profile" will be the owner of a job, specify **This Account** (and not **Local System Account**) in the CONTROL-M/Agent Service Definition window.
- The owner must have access rights to the network and the **Log on as a batch job** user right.

For information about how to assign user rights, see page 28.

NOTE

To eliminate the need to assign user rights to every job owner on every Microsoft Windows computer running CONTROL-M/Agent, BMC Software recommends that you define a domain-level group for all job owners. You can name this group CONTROL-M Job Owners. Assign network access rights and the **Logon as a batch job** user right to this group.

How to assign user rights to agent users

You can use this procedure to assign the user rights listed on page 28 to each user who needs them.

Assigning user rights

1. Log on to the CONTROL-M/Agent computer as a local administrator.
2. Choose **Start => Settings =>Control Panel=> Administrative Tools => Local Security Policy**.
3. In the displayed tree structure, select **Local Policies**.
4. In the displayed panel, double-click **User Rights Assignments** to display the list of user rights.
5. Double-click the user right you want to assign. The **Local Security Policy Settings** window for that user right is displayed.

A If the user who should have the selected user right is not listed in this window, click **Add**.

B In the bottom panel, enter the **<domain>\<user_name>** of the user and click **OK**.

C When the specified user is displayed in the lower panel, click **OK** again.

6 Restart the agent services to make the user rights effective.

Verifying communication with the server

BMC Software recommends that you verify the ability of the agent computer to communicate with the primary Server computer and with all other authorized Server host computers.

Generating the Communication Diagnostic Report

CONTROL-M/Agent includes a diagnostic program that checks parameters and environmental conditions relevant to communication between the agent and server computers. This program is typically used at the request of Technical Support to determine the cause of a communication problem.

To generate the Communication Diagnostic Report

1 Navigate to the directory in which CONTROL-M/Agent is installed.

2 Enter the `ag_diag_comm` command. The CONTROL-M/Agent Communication Diagnostic Report is displayed. If the user is not the administrator, the `ag_diag_comm` command cannot display all the details.

```
CONTROL-M/Agent Communication Diagnostic Report
-----------------------------------------------

 Agent User Name                : ag620
 Agent Directory                : /home/ag620/ctm
 Agent Platform Architecture    : AIX
 Agent Version                  : 6.3.01.000
 Agent Host Name                : appsrv002
 Server Host Name               : sunsrv001
 Authorized Servers Host Names  : sunsrv001
 Server-to-Agent Port Number    : 7006
 Agent-to-Server Port Number    : 7005
```

```
Server-Agent Protocol Version  : 06
Server-Agent Comm. Protocol    : TCP
Server-Agent Connection mode   : Transient
Unix Ping to Server Platform   : Succeeded
Agent Ping to CONTROL-M/Server : Succeeded

Agent processes status
======================
Agent Router                   : Not running
Agent Listener                 : Running (42762)
Agent Tracker                  : Running (51208)
```

Sending a shout message to an e-mail address

The CONTROL-M Shout facility sends messages to recipients based on specified logical destinations. For more information, see the *CONTROL-M/Server for UNIX and Microsoft Windows Administrator Guide*.

CONTROL-M / Server can request CONTROL-M / Agent to send a message and a SYSOUT to an e-mail destination. If the destination is not a standard e-mail address, CONTROL-M / Agent will try to resolve the destination to an e-mail address using the **Address Book** of the default Windows Messaging Component (for example, Microsoft Outlook).

Language capabilities

Starting with version 6.3.01, CONTROL-M supports two foreign language modes:

- Latin1 (German, French, and Spanish)
- CJK (Chinese, Japanese, Korean)

Information about CONTROL-M / Agent language support is found in the *CONTROL-M Language and Customization Guide*.

Chapter 3

Writing scripts for UNIX

When writing a shell script to be run as a CONTROL-M/Server job on an agent computer, the following factors must be considered:

- Specification of the shell type under which the script will run.
- Run-time environmental factors affecting execution of the script.
- Usage of the **On Statement/Code** job processing parameter.

BMC Software recommends that you run each script manually to validate the script syntax before running the script under CONTROL-M/Server.

NOTE

The command line of command type jobs must be in Bourne shell syntax only.

Specifying the shell type

To enable CONTROL-M to recognize the script shell type, specify the shell path (as listed in Table 4) on the first line of the script:

```
#! <shell path>
```

Table 4 Shell paths

Shell type	Shell path	Default switch	Other switches
Bourne	**/bin/sh**	-x	-v/n
Korn	**/bin/ksh**	-x	-v/n
csh	**/bin/csh**	-v	
tcsh	**/bin/tcsh**	-v	

NOTE

BMC Software recommends that you not add a switch to the above syntax. To toggle the switches for the scripts, use the OS CM tab of the Agent Configuration utility. For more information, see "OS CM tab (UNIX)" on page 57.

Table 5 describes the affect Shell parameter switches have on CONTROL-M / Agent processing.

Table 5 Shell parameters

Parameter	Description
-x	This parameter causes CONTROL-M / Agent to submit the script as is. The script runs under the specified shell and prints commands and related arguments as they are executed. **Note**: In the SYSOUT file, the command arguments contain the value of the variable and not the variable name. Each command is prefixed by the '**+**' sign. This sign is later used during an On statement post-processing phase of the jobs output to distinguish between the different commands and their output.
-v	This parameter causes CONTROL-M / Agent to parse the original script to a temporary script. The script commands are appended with an identifying string. This temporary script is then executed, where the *-v* switch causes the shell to print each command before its output. The added identifying string is later used during an On statement post-processing phase of the job's output to distinguish between commands and their output.
n	This CONTROL-M / Agent-specific flag is used to indicate that the script should be executed as is and no commands will be included in the job's output. As a result no On-statement processing is possible.
For more information about the different flags, see the example on page 33.	

NOTE

Arguments specified after the shell name are ignored by CONTROL-M / Agent with the following exception: **-x** is supported when running a script under the Bourne shell or Korn shell. If **-x** is specified as an argument after the shell name, it overrides any option set in the CTM_PRM_SH_FLAGS or CTM_PRM_KSH_FLAGS parameters.

Example

The following script uses the app, dbadmin, and stx111 parameters. The app parameter sets an environment variable. The script uses the dbadmin and stx111 parameters to call a utility that performs an action. The output of the job varies depending on the shell flag.

EXAMPLE

```
#!/bin/sh
DBNAME=$1
export DBNAME
dbrefresh -U $2 -P $3
exit $?
```

- If the *-x* flag was set when running the sample script, the job produces the following output.

```
DBNAME=app
+ export DBNAME
+ dbrefresh -U dbadmin -P stx111
DB refreshed
+ exit 0
```

- If the *-v* flag was set when running the sample script, the job produces the following output.

```
#! /bin/sh -v
CTM_RSVD=
CTM_RSVD_START=
CTM_RSVD_END=
CTM0='/home2/ag620/refreshDB.sh'
CTM00=$0
DBNAME=$1 $CTM_RSVD
export DBNAME $CTM_RSVD
dbrefresh -U $2 -P $3 $CTM_RSVD
DB refreshed
exit $? $CTM_RSVD
```

- If the *n* flag was set when running the sample script, the job produces the following output.

```
DB refreshed
```

Support for REXX-Language scripts

On certain computers, job scripts to be run under CONTROL-M can be written using the REXX shell language. REXX job scripts are supported on the following computers: AIX, SunOS, and Solaris.

To activate a REXX script, the REXX product must first be installed on the agent computer. The first line of the REXX script must specify the full path under which REXX is installed.

EXAMPLE

```
#!/usr/local/bin/rxx
```

Run-time environment

CONTROL-M runs a job script under the environment specified for the job owner (that is, the user specified in the **Owner** parameter). The environment affects these factors in the execution of the script:

- User log on process
- Shell script startup process
- Working directory

Each of these factors is described below.

User log on process

As jobs are submitted for execution, CONTROL-M / Agent logs on as the user and executes the job (the shell script) using the following command:

```
su - <owner> -c <script name>
```

During the logon process, the user environment is set according to the shell type specified in / **etc/passwd**.

Shell script startup process

The startup process for running the script depends upon the type of shell under which the script will run.

- When a csh or tcsh script is run, the **.cshrc** file of the job owner is executed as part of the startup process for the script.
- For all other shell types, the **.profile** file of the job owner is executed as part of the startup process for the script.

NOTE

The **.login** file is not executed as part of the startup process.

When CONTROL-M executes job scripts, there is no terminal associated with the job. Therefore, do not use commands in a script that query terminal characteristics or take input from a terminal.

The shell script startup process sets the environment variables that will be available when the script is run. The **#!** statement (see "Specifying the shell type" on page 32) indicates the shell under which the script is intended to run.

Working directory

The working directory at the time the script runs is initially set to the home directory of the job owner (the home directory for each user is set by the UNIX administrator in **/etc/passwd**).

When writing scripts that access files, the file name in the script should be specified with a full path or with a path relative to the home directory of the job owner.

On Statement/Code parameter

The following items describe how the **On Statement/Code** job processing parameter interprets script lines.

- **Type of Script Statement**
  Depending on the shell used, CONTROL-M/Agent does not process certain types of script statements for comparison with the text specified in the **Stmt** subparameter of the **On Statement/Code** parameter. Therefore, text contained in these script statements should not be specified in the **Stmt** subparameter:

  — For a Bourne shell, text in **if, for, while**, and **case** statements.

  — For a csh shell, text in **if** statements.

EXAMPLE

No part of the following script line should be used in the **Stmt** subparameter of the **On Statement/Code** parameter:

```
if [ 'baseline' - eq 0 ]; then
```

- **Continuation Lines**
  CONTROL-M/Agent does not process continuation lines for comparison with text specified in the **Stmt** subparameter of the **On Statement/Code** parameter. Therefore, text on a continuation line in a script should not be specified in the **Stmt** subparameter.

- **Length of Script Statement**
  CONTROL-M/Agent only processes the first 132 characters of a script statement for comparison with the text specified in the **Stmt** subparameter of the **On Statement/Code** parameter. Therefore, text that occurs after the first 132 characters of a script statement should not be specified in the **Stmt** subparameter.

- **HERE Documents**
  The term HERE document refers to lines of text in a script that are passed to a command as input, but are not passed to the shell. The current version of CONTROL-M/Agent does not support the **On Statement/Code** job processing parameter for HERE documents.

EXAMPLE

In the following script, **line 1** and **line 2** of a **HERE** document are passed to the specified **cat** command:

```
cat > /tmp/junk << EOF_EOF
line 1
line 2
EOF_EOF
echo "DONE"
```

For more information about the **On Statement/Code** parameter, see the *CONTROL-M Job Parameter and Variable Reference Guide*. Job processing parameters are described in the *CONTROL-M/Enterprise Manager User Guide*.

Utilization of Exit Codes by CONTROL-M/Server

You can cause CONTROL-M/Server to distinguish between different exit codes by using the following expression in the **Code** subparameter of the **On Statement/Code** job processing parameter:

```
COMPSTAT=<value>
```

<value> is the exit code of the script.

EXAMPLE

Assume that a script exits with an exit code of 5.
This condition can be detected by defining the following **On Statement/Code** parameters:

```
Stmt: *
Code: COMPSTAT=5
```

Use of the $0 Reserved Variable

The **$0** reserved variable can be used in a script to retrieve the name of the script. This variable is automatically replaced by a file name before the script is run. When a script runs as a CONTROL-M/Server job using the *–v* flag (see Specifying the shell type), it is parsed into a temporary script so any reference to $0 in the script is resolved to the temporary script name. The name of the original script is saved in the CTM0 variable. This differentiates between a script run from the command line run and a script run from a CONTROL-M/Server job.

To resolve this problem you need to set the Translate_$0 flag using the OS tab of the agent Configuration utility. For more information, see "OS CM tab (UNIX)" on page 57.

Setting the flag causes CONTROL-M/Agent to replace any occurrence of $0 in the original script with $CTM0 when it parses the original script to the temporary script. This will restore the original functionality of the script as if it ran from the command line.

The following example shows the **dollar0.sh** script, which is supposed to print out the script name.

EXAMPLE

```
#!/bin/sh
echo $0
```

- When the script runs as part of a CONTROL-M/Server job using the -v flag, the name of the temporary script is printed.

```
#! /bin/sh -v
CTM_RSVD=
CTM_RSVD_START=
CTM_RSVD_END=
CTM0='/home/ag620/dollar0.sh'
CTM00=$0
echo $0 $CTM_RSVD
/tmp/ctm/CM_SH.11939
```

- When the script runs in a CONTROL-M/Server job using the *-v* flag *and* the Translate_$0 flag is set, the name of the original script is printed.

```
#! /bin/sh -v
CTM_RSVD=
CTM_RSVD_START=
CTM_RSVD_END=
CTM0='/home/ag620/dollar0.sh'
CTM00=$0
echo $CTM0 $CTM_RSVD
/home/ag6220/dollar0.sh
```

Chapter 4

Writing scripts for Windows

When operating in the Microsoft Windows environment, CONTROL-M/Agent supports the following types of job scripts:

- DOS batch files (suffix**.bat**)
- REXX-language scripts (suffix**.cmd**)

CONTROL-M/Agent can use the **On Statement/Code** job processing parameters to perform post-processing analysis of the SYSOUT of jobs that are submitted by using these scripts.

The following topics are discussed in this chapter:

Basic guidelines

Scripts analyzed by CONTROL-M / Server as part of the post-processing of a job should comply with the following requirements:

- Begin the script with the **echo on** command. This ensures that job script statements will be written to the SYSOUT file.
- End each prompt with a **>** or **]** character. These characters and embedded spaces should not be used inside the prompt text string.

On Statement/Code parameter

The following items describe how the **On Statement/Code** job processing parameter interprets script lines:

- **Analysis of the Sysout for On Statement/Code**

  Text in a SYSOUT file that follows a **>** prompt or **]** prompt is treated by CONTROL-M / Server as part of the job script. All other text is treated as part of the operating system response.

  When specifying an **On Statement/Code** statement (format 1) in a job processing definition, place text that follows either of these prompts in the **Stmt** parameter. Place other text in the **Code** parameter.

- **Continuation Lines**

  CONTROL-M / Server does not process continuation lines for comparison with text in a **Stmt** subparameter. Therefore, do not specify script continuation line text in the **Stmt** subparameter.

- **Length of script statement**

  CONTROL-M / Server compares the first 512 characters of a script statement with the text in subparameter **Stmt**. Text after the first 512 characters of a script statement should not be in subparameter **Stmt**.

  The maximum length of the **On Code** parameter is 1024 characters.

For more information about the **On Statement/Code** parameter, see Chapter 7 of the *CONTROL-M Job Parameter and Variable Reference Guide*. Job processing parameters are described in Chapter 5 of the *CONTROL-M/Enterprise Manager User Guide*.

Utilization of Exit Codes by CONTROL-M/Server

Both DOS **.bat** scripts and REXX **.cmd** scripts can return an exit code to CONTROL-M / Server upon completion. The **_exit** utility described below is used by **.bat** scripts.

CONTROL-M / Server can distinguish between exit codes by using the following expression in the **Code** subparameter of the **On Statement/Code** job processing parameter:

```
COMPSTAT=<value>
```

Example

In this example, a REXX script exits with an exit code of 5, as shown below:

```
exit  5
```

This condition can be detected by defining the following **On Statement/Code** parameter:

```
Stmt: *
Code: COMPSTAT=5
```

Script utilities

The _exit and _sleep script utilities can be accessed from within job scripts. These utilities are located in the **<CONTRL-M/Agent>\EXE** directory under the Product directory, for example:

```
c:\Program Files\BMC Software\CONTRL-M Agent\<agent name>\EXE
```

If this directory is not defined as part of the operating system search path, specify the full path when using one of these utilities.

_exit

This utility is similar to the UNIX **exit** built-in shell function.

The utility is located in the **<CONTRL-M/Agent>\EXE** directory path that was created during the installation procedure.

Format

```
_exit [<exit code>]
```

The variable <exit code> is any whole number Š 0 Default: **0**

The program exits with %errorlevel% = <exit code>

Examples

_exit 0 in a script causes the job to end with **%errorlevel% 0**.

```
ctmcreate -tasktype command -cmdline "_exit 0"
```

_exit 1 in a script causes the job to end with **%errorlevel% 1**.

```
ctmcreate -tasktype command -cmdline "_exit 1"
```

_sleep

This utility is similar to the UNIX **sleep** built-in shell function.

The utility is located in the **<CONTRL-M/Agent>\EXE** directory path that was created during the installation procedure.

Format

```
"... _sleep" <seconds>
```

The <seconds> variable is any whole integer number Š 0

NOTE

If _sleep is specified, you must specify a whole integer number.

Example

Suspend execution of the script for 5 seconds.

ctmcreate -tasktype command -cmdline "_sleep 5"

Translating DOS files and REXX scripts to UNC

The **CTMBAT2UNC** utility translates DOS batch files (**.bat**) and REXX-language (**.cmd**) scripts containing mapped path names into scripts that use Universal Naming Convention (UNC) equivalents, to reference remote disk resources. These translated scripts enable CONTROL-M / Agent to execute multiple scripts simultaneously. The owners of the jobs do not have to be logged on to provide the drive mappings for the scripts.

The CTMBAT2UNC utility can be invoked using the following command:

ctmbat2unc.exe <batch_file_to_translate> <output_file_name>

Table 6 describes the CTMBAT2UNC utility parameters.

Table 6 CTMBAT2UNC Utility Parameters

Item	Description
<batch_file_to_translate>	Original **.bat** or **.cmd** script
<output_file_name>	New script after translation

Example

Two job owners, A and B, are executing ScriptA.bat and ScriptB.bat, respectively. Owner A has drive M mapped to **\\nt-A\share**. Owner B has drive M mapped to **\\nt-B\share**.

Table 7 describes these scripts before and after executing the **CTMBAT2UNC** utility.

Table 7 Scripts before and after running CTMBAT2UNC

Owner	Original script	Translated script
A	@echo off dir M:\jobs	@echo off REM Following line was changed by CTMBAT2UNC dir \\nt-A\share\jobs
B	@echo off dir M:\jobs	@echo off REM Following line was changed by CTMBAT2UNC dir \\nt-B\share\jobs

As shown above, every line changed by the **CTMBAT2UNC** utility is marked by a REM comment inserted before the translated line.

NOTE

Under the current version of Microsoft Windows, command interpreters do not change a current directory to a UNC path (for example, `cd \\nt-A\share\jobs` will not be executed).

BMC Software recommends that you review the translated script after invoking the ctmbat2unc utility.

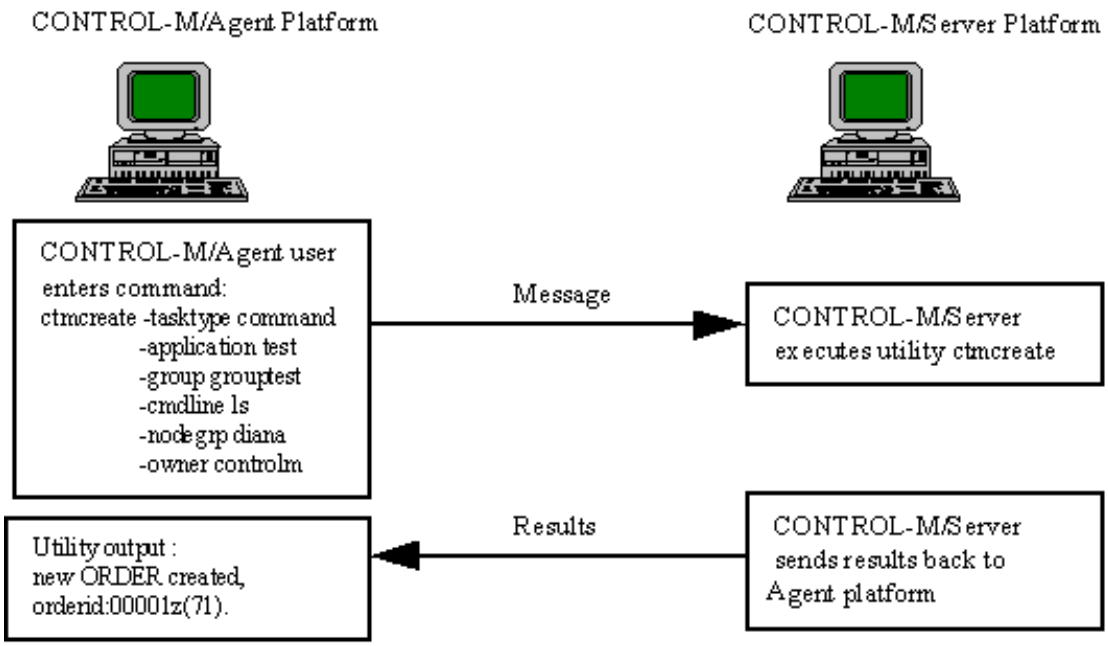
Utilities

The utilities in Table 9 on page 51 can be invoked on an agent computer by the user or by a batch job running on the agent computer.

Some of these utilities are executed by CONTROL-M / Server. Their output is sent to the agent computer. These utilities are described in the utilities chapter of the *CONTROL-M/Server for UNIX and Microsoft Windows Administrator Guide*. Their processing workflow is illustrated in Figure 2.

Most utilities that create a job in the CONTROL-M / Server Active Jobs file are interactive when invoked from the Server computer, but not interactive when invoked from the agent computer. When invoked from the agent, they must be invoked with all the required parameters.

Figure 2 CONTROL-M/Server utility workflow



NOTE

If the primary CONTROL-M/Server does not respond to a CONTROL-M/Agent request to execute a utility (other than ag_ping), the request is automatically redirected to the first non-primary Server listed in the Authorized CONTROL-M/Server Hosts parameter. If the redirection is successful, that agent continues to work with the replacement server.

Timeout intervals

The Agent-to-Server and Server-to-Agent communication timeout intervals are described on page 53. If the agent requests a utility that runs on the server, or the server requests the agent to order a job, and there is no response within the timeout interval, the requested action will fail.

You can increase these timeout intervals by using the Advanced window of the agent configuration utility described on page 55. However, increasing these timeout intervals tends to reduce CONTROL-M/Agent performance.

Specifying utility parameters

When a utility is invoked from the command line, a maximum of 1000 characters can be entered on the command line. The command used to invoke a CONTROL-M/Server utility is normally specified with all relevant parameters.

Table 8 lists utilities that enable you to place utility parameters in an input file. The **-input_file** parameter identifies the file that contains parameters for the utility. In this file, each parameter and its values (if any) are on a separate line with the same syntax they would have on a command line.

Using the **-input_file** parameter enables you to

- prepare and save files of utility parameters that can be reused
- exceed the number of characters allowed in the command line

Table 8 Utilities that support the -input_file parameter

ctmcontb	ctmkilljob	ctmshout
ctmcreate	ctmorder	ctmvar
ctmdefine	ctmstvar	ecaqrtab

Example

Assume that you have to specify the ctmcreate utility. In this case the parameters with their values exceed the 1000 character limit. You can specify the parameters and values in an input file, as follows:

ctmcreate -input_file <*fileName*>

The <*fileName*> variable is the name of a file that contains the ctmcreate parameters. For example:

-tasktype command
-cmdline ls

Format for command lines with spaces

If a command invoked from the agent computer contains embedded spaces, add **\"** after the first quote at the beginning of the command, *and* add **\"** at the end of the command (but prior to any parameters).

Example

```
"\"d:\program files\bmc software\control-m agent\exe\_sleep\" 200"
```

Utility parameter syntax (*UNIX*)

Some utilities require special formatting for transmission to the Server computer. See "Special utility parameter formats"below.

Values for utility parameters must not contain the apostrophe or single quote character.

When invoking these utilities from CONTROL-M/Agent, the presence of a special character in the argument values may cause problems. The following command contains a back-slash before the string DELETEME:

```
ctmcreate -tasktype COMMAND -jobname servertest -cmdline "ctmvar -action set -var
'%%#\DELETEME' -varexpr to_be_deleted"
```

When this command is invoked from CONTROL-M/Agent, the back-slash before DELETEME may be “eaten” by the shell. To avoid this problem, add a back-slash before the special character that causes the problem (in this case, the original back-slash).

When invoking the ctmcreate utility or ctmdefine utility from CONTROL-M/Agent with a **date_ref** of **$$$$**, put a back-slash before each **$** as shown here:

```
ctmcreate -tasktype command -cmdline ls -incond a '\$\$\$\$' AND ...
```

Special utility parameter formats

Commands invoked from UNIX agent computers are embedded in double quotes when sent to the Server computer. Therefore, use single quotes for command elements that must be within quotation marks. For example:

```
ctmcreate ...-cmdline "ls -l '$HOME'"
```

For a description of syntax rules when invoking the ctmcreate utility, see “ctmcreate” in the utilities chapter of the *CONTROL-M/Server for UNIX and Microsoft Windows Administrator Guide*.

Directing output from utilities

Some utilities generate reports that can be directed to a file. These utilities have <Output_ parameters>.

- If output parameters are specified, the utility output is directed to a file on the Server computer.
- If output parameters are not specified, the output is routed to the default output device, for example, the logical name of a disk.

Output can be redirected to the agent computer by specifying a full path name of the file after the redirection (>) character.

Enabling other users to run agent utilities (*UNIX*)

Users other than the CONTROL-M/Agent user

To enable users other than the CONTROL-M / Agent user to invoke these utilities from CONTROL-M / Agent, add the following environment variables to **.cshrc** or **.profile**:

Add to .cshrc

```
set path = ( ${path} <agentHome>/ctm/scripts <agentHome>/ctm/exe )
setenv CONTROLM "<agentHome>/ctm"
if ( ${?LD_LIBRARY_PATH} ) then
setenv LD_LIBRARY_PATH ${LD_LIBRARY_PATH}:<agentHome>/ctm/exe
else
setenv LD_LIBRARY_PATH "<agentHome>/ctm/exe"
endif
```

Add to .profile

```
CONTROLM=<agentHome>/ctm
export CONTROLM
PATH=$PATH:<agentHome>/ctm/exe:<agentHome>/ctm/scripts
export PATH
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:<agentHome>/ctm/exe
export LD_LIBRARY_PATH
```

<*agentHome*> stands for the CONTROL-M / Agent account home directory.

Invoking a utility on a computer with Multiple Agents (*Windows*)

All CONTROL-M/Agent utilities now support the -agent <agent name> parameter. The variable <agent name> represents the name of the CONTROL-M/Agent specified during the installation procedure.

Where multiple CONTROL-M/Agents reside on a computer, the -agent parameter determines which CONTROL-M/Agent will manage the utility. If the ctmagcfg or ag_diag_comm utilities are run without specifying the -agent parameter, the user is prompted to select the CONTROL-M/Agent. For all other utilities, if the -agent parameter is not specified, the default CONTROL-M/Agent is used. For more information, see "Communicating With a Specific CONTROL-M/Agent" on page 19.

Example

Assume a computer has two Agents, **Default** and **Saturn**. To add a user to **Default**, use the following command:

ctmpwd -action add -user user2 -password 123456 -agent Default

-or-

ctmpwd -action add -user user2 -password 123456

To add a user to **Saturn**, use the following command:

ctmpwd -action add -user saturn_user2 -password 123456 -agent Saturn

Utility descriptions

Table 9 lists utilities that can be invoked from CONTROL-M/Agent. Some of these utilities can use the **-input_file** parameter. For more information, see "Specifying utility parameters" on page 46.

Utilities that are not described in this chapter are described in the utilities chapter of the *CONTROL-M/Server for UNIX and Microsoft Windows Administrator Guide*.

NOTE

Agent utilities run on a Windows computer by a user other than the administrator may not have access to certain data or functions.

Table 9 CONTROL-M/Agent utilities (part 1 of 2)

Utility	Computer	Description
Utilities that run on the agent computer		
_exit	Windows	Sets the completion status for a job run from a **.bat** file. Exit code **0** is equivalent to **Ended OK**. Any other exit code is equivalent to **Ended NOTOK**. For more data, see page 42.
_sleep	Windows	Determines the sleep time for all CONTROL-M/Server processes or for a specific process. For more information, see page 42.
ag_diag_comm	UNIX and Windows	Agent communication diagnostic utility.
ag_ping	UNIX and Windows	Verifies that CONTROL-M/Server is active on the Server computer that is connected to the agent computer. For more information, see "ag_ping utility" on page 52.
ctmag	UNIX and WIndows	Interactively configures parameters in the config table. For more information, see "Agent configuration utility" on page 53.
ctmfw	UNIX and Windows	Detects completion of file transfer activity by checking file size. Signals when desired file size is achieved. Can be invoked from the **<CONTROL-M/Agent>/exe** directory. For more information, see page 61**.**
ctmpwd	Windows	Maintains CONTROL-M user and password information. For more information, see page 74. (This utility replaces the **ctmcpt** utility in earlier versions.)
ctmunixcfg	UNIX	Enables you to view and modify most of the configuration parameters in the **OS.dat** file. For more information, see the "Agent configuration utility" on page 53.
ctmwincfg	Windows	Enables you to view and modify Control Module for Windows configuration parameters. For more information, see page 58.
shagent	UNIX	Shows if an agent and Tracker are running.
		Note: This utility can be invoked only from CONTROL-M/Agent.
Utilities that run on the server computer		
ctmcontb	UNIX and Windows	Performs operations on the Prerequisite Conditions table.
ctmcreate	UNIX and Windows	Creates a job in the CONTROL-M/Server Active Jobs file.
ctmdefine	UNIX and Windows	Defines a job in the CONTROL-M/Server database.
ctmgrpdef	UNIX and Windows	Creates a definition for a new Group Scheduling table.
ctmkilljob	UNIX and Windows	Terminates a CONTROL-M/Server job and its associated processes.
ctmloadset	UNIX and Windows	Updates a Quantitative resource in the Resources table. For more information, see the *CONTROL-M/Server for UNIX and Microsoft Windows Administrator Guide*.
ctmnodegrp	UNIX and Windows	View and maintain node groups. For more information, see the utilities chapter of the *CONTROL-M/Server for UNIX and Microsoft Windows Administrator Guide*.

Table 9 CONTROL-M/Agent utilities (part 2 of 2)

Utility	Computer	Description
ctmorder	UNIX and Windows	Orders or forces one or more jobs from a Scheduling table contained in the CONTROL-M/Server database.
		Note: When this utility is invoked from CONTROL-M/Server, parameters can be specified in a fixed order without parameter tags or in any order with tags. When invoked from a CONTROL-M/Agent computer, each parameter must be preceded by its tag.
ctmpsm	UNIX and Windows	Displays the CONTROL-M Active jobs file (AJF). For more information, see the *CONTROL-M/Server for UNIX and Microsoft Windows Administrator Guide*.
ctmshout	UNIX and Windows	Issues a shout message to an indicated destination.
ctmstvar	UNIX and Windows	Displays the current value of an AutoEdit variable or function.
		Note: When this utility is invoked from CONTROL-M/Server, parameters can be specified in a fixed order without parameter tags or in any order with tags. When invoked from a CONTROL-M/Agent computer, each parameter must be preceded by its tag.
ctmudly	UNIX and Windows	Orders jobs for a specific User Daily name.
ctmvar	UNIX and Windows	Maintains AutoEdit variables.
		Note: The value of parameter **-filename** is the full path and name of a file that is accessible to CONTROL-M/Server.
ecactltb	UNIX and Windows	Lists the status of each Control resource in the Resources table.
ecaqrtab	UNIX and Windows	Performs operations on quantitative resources in the Resources table. Syntax when invoked from the agent: `ecaqrtab LIST "*" [-OUTPUT <Output>]`

Utilities that are not described on the following pages are executed in CONTROL-M/Server and are described in the utility chapter of the *CONTROL-M/Server for UNIX and Microsoft Windows Administrator Guide*.

If users other than the administrator invokes these utilities from CONTROL-M/Agent, the utilities will run but not all the features and data will be available. [Windows].

ag_ping utility

This utility verifies that CONTROL-M/Server is active on the Server computer connected to the agent computer. From the operating system prompt, specify the following command:

```
ag_ping
```

The utility attempts to communicate with CONTROL-M/Server and indicates whether the attempt succeeded or failed. If the attempt succeeds, you will receive the message:

```
Output:
Server is alive.
Result: Success.
```

Agent configuration utility

The Agent Configuration (ctmag) utility is a Java application used to maintain CONTROL-M/Agent configuration parameters, and to view and modify most of the operating system parameters. If the user running the utility is not an administrator, changes made to agent configuration parameters will not be saved.

NOTE

For information about running the Agent Configuration utilities as command_line utilities, see "Command_line utilities" on page 60.

There are additional tabs that represent each of the Control Modules (CMs) installed on the agent.

This utility is located at `<installation_dir>\EXE\ctmag` and can be run from the command prompt or, on Windows using **Start > Programs > CONTROL-M/Agent 6.3.01**.

NOTE

If more than one agent is installed, use the `ctmag <agent name>` command when opening the Agent Configuration utility from the command line.

Agent tab

Table 10 lists the parameters displayed in the Agent tab and their descriptions.

NOTE

Right-clicking on a field label and selecting **About it** provides context sensitive help for that field.

Table 10 ctmag utility parameters

Parameter	Attributes
Agent-to-Server Port	CONTROL-M/Server and CONTROL-M/Agent communicate by means of two TCP/IP ports. This parameter specifies the port in the Server computer that receives data from the agent computer. The value specified must match the value assigned to the Agent-to-Server Port Number parameter on the Server computer. Verify that the specified port number is not used for any other purpose in the agent. Specify a numeric value from 1025 to 65535, inclusive. Default: **7005**
Server-to-Agent Port	CONTROL-M/Server and CONTROL-M/Agent communicate by means of two TCP/IP ports. This parameter specifies the port in the Server computer that sends data to the agent computer. The value specified must match the value assigned to the Server-to-Agent Port Number parameter on the Server computer. Verify that the specified port number is not used for any other purpose in the agent computer. Specify a numeric value from 1025 to 65535, inclusive. Default: **7006**
Primary CONTROL-M/Server Host	Host computer for the CONTROL-M/Server that manages this agent. Type the name of the primary CONTROL-M/Server host in the field box or select a host name from the list box. Default: Computer on which the installation was made.
	Note: Do not use a numeric IP address, such as **173.19.6.14,** to specify the name of the server.
Authorized CONTROL-M/Server Host	Names of all CONTROL-M/Servers authorized to manage this agent (including the primary Server). Specify the host names separated with the "|" symbol. Do not include spaces between host names. The host name of the primary CONTROL-M/Server is automatically included in this field. Default: Name of the primary CONTROL-M/Server host.
	Note: Specify names, such as UNIX_HOSTNAME. Do not specify IP addresses.
Diagnostic Level	Flag that indicates whether to generate diagnostic messages. Valid values: **0**–**4**. Level **0** generates no diagnostics. Level **4** generates maximum diagnostics. This parameter can only be changed after completing the installation.
Communication Trace	Flag that indicates whether to debug communications between CONTROL-M/Agent and CONTROL-M/Server. Valid values: **0** = no. **1** = yes. Default: **0** This parameter can only be changed after completing the installation.

NOTE

You can also access the ACU application by clicking on the **Agent Diagnostic** button located at the bottom of the Agent tab. For more information about ACU, see "Agent check utility" on page 76.

Changes to the Diagnostic Level parameter are saved and take effect immediately. If you change any other parameter, the configuration parameters are saved and take effect after CONTROL-M/Agent is restarted.

Agent tab (Advanced)

Table 11 lists the parameters found under the Advanced Agent window of the Agent tab and their descriptions.

Table 11 ctmag (advanced) utility parameters (part 1 of 2)

Parameter	Attributes
Allow_Comm_Init	Determines whether the Agent Router can initiate a session with the server. Valid values: Y/N. Default: **Y** If the server sits behind a firewall, this parameter should be set to **N**. For more information about connection models, see "Agent to Server connection models" on page 20.
AutoEdit Inline	Flag that indicates whether AutoEdit variables defined in a CONTROL-M/Server job are set as Environment variables in the user job environment. This parameter can only be changed after completing the installation.Valid values: ■ **Checked** - AutoEdit variables are set. Default. ■ **Cleared** - AutoEdit variables are not set.
CTMS Address Mode	If this parameter is set to **IP**, the IP address instead of the host name is saved in CTMS_HOSTNAME. Use this parameter when CONTROL-M runs on a computer with more than one network card.
Daily Log File Enabled	Indicates whether the **ctmag_<year><month><day>.log** file is generated (**Y**) or not (**N**). Default: **Y**
Days To Retain Log Files	Number of days that agent proclog files are retained. After this period, all agent proclog files are deleted by the New Day procedure. Default: **1**
Foreign Language Support	Determines the regional settings supported by CONTROL-M/Agent. Valid values are: ■ Latin-1 (default) ■ CJK For more information, see the *CONTROL-M Language and Customization guide*.
Listen to Network Interface	Indicates the IP address or host name on which the agent listens for requests from CONTROL-M/Server. The default value is *ANY, meaning that CONTROL-M/Agent listens on all interfaces.
Locale (UNIX only)	Determines the regional settings used by the CONTROL-M/Agent account. This parameter is only valid when the CONTROL-M/Agent language is defined as Latin-1. For more information, see the *CONTROL-M Language and Customization guide*.
Logical Agent Name	The name of the agent. The default value is the name of the computer on which you installed CONTROL-M/Agent.

Table 11 ctmag (advanced) utility parameters (part 2 of 2)

Parameter	Attributes
Persistent Connection	Indicates the connection model between the agent and server. Valid values: Y / N. Default: **N** When set to Y, the server and agent maintain a constant connection allowing traffic from all agent processes to reach the server, even if the Server is behind a firewall. For more information about connection models, see "Agent to Server connection models" on page 20.
Protocol version (Windows only)	Server-Agent communication protocol version. Default: **07**
SSL (Windows only)	SSL support for CONTROL-M / Agent for Microsoft Windows. When SSL support is implemented in CONTROL-M / Server it makes a one-time request to set this parameter automatically for each associated agent. This request requires 2 to 5 minutes for each agent. Setting this parameter manually can save time if a large number of Agents work with CONTROL-M / Server. For more information about setting this parameter manually, see "Agent configuration utility" on page 53.
TCP/IP Timeout	Communication timeout in seconds. Specify a numeric value greater than or equal to zero. Default: UNIX **120**; Windows **60**
Timeout for Agent utilities	Maximum time (in seconds) the agent waits after sending a request to CONTROL-M / Server. This timeout interval should be longer than the TCP / IP Timeout. Recommended value and default: UNIX **120**; Windows **600**
Tracker Polling Interval	Time in seconds that the Tracker waits after starting the job status checking process before re-starting that process. This parameter can only be changed after completing the installation. Default: UNIX **120**; Windows **60**
Tracker Port	Number of the port for sending messages to the Tracker process when jobs status changes. The tracker event port enables CONTROL-M / Agent to receive updates regarding job status from all Control Modules associated with the current CONTROL-M / Agent. Default: **7009**

Table 12 Supported locale settings

Language	AIX and Solaris	HP-UX
English (USA)	en_US.ISO8859-1	en_US.iso88591
English (British)	en_GB.ISO8859-1	en_GB.iso88591
German	de_DE.ISO8859-1	de_DE.iso88591
French	fr_FR.ISO8859-1	fr_FR.iso88591
Spanish	es_ES.ISO8859-1	es_ES.iso88591

OS CM tab (*UNIX*)

Table 13 lists the parameters displayed in the OS CM (*UNIX*) tab (ctmag configuration utility) and their descriptions.

To prevent Korn and Bourne shell script output from being included in system output (SYSOUT), set Korn Shell Flags and Bourne Shell Flags to **n**.

Table 13 OS CM (*UNIX*) tab parameters

Parameter	Description
Printer Name	Name of the printer. Default: **lpr**
Sysout Name	Determines the prefix for the Sysout file name. Valid values: ■ **MEMNAME** – the Sysout file prefix is the MEMNAME of the job. Default. ■ **JOBNAME** – the Sysout file prefix is the JOBNAME of the job.
Korn Shell Flags	Indicates the shell flag that will be used to run the job script. For more information, see "Specifying the shell type" on page 32. Valid values: **-x**, **-v, n**. Default: **-x**
Bourne Shell Flags	Indicates the shell flag that will be used to run the job script. For more information, see "Specifying the shell type" on page 32. Valid values: **-x**, **-v**, **n**. Default: **-x**
Temporary scripts saving	Indicates if temporary scripts generated from jobs are deleted at the end of job execution. Valid values: YES or NO. ■ YES – the temporary scripts are not deleted ■ NO – the temporary scripts generated from jobs are deleted at the end of job execution Default: **NO**
Temporary Scripts Directory	The default path for saving the temporary scripts.
Replace $0 by filename	■ If set to **Y**, reserved variable **$0** specifies whether instances of $0 in the job script should be replaced before the script is run. Default. ■ If set to **N**, this functionality is disabled. Default. For more information, see "Use of the $0 Reserved Variable" on page 37.
CJK encoding	Determines the CJK encoding used by CONTROL-M/Agent to run jobs. For more information, see the *CONTROL-M Language and Customization guide*.

Click **SMTP Parameters** to display the SMTP Parameters tab.

SMTP Parameters tab

Table 14 lists the parameters displayed in the SMTP Parameters tab and their descriptions.

Table 14 SMTP tab parameters

Parameter	Description
SMTP Server <Relay> Name	The name of the SMTP server
Port Number	The port number on which the SMTP server communicates. Default: 25
Sender Email	The e-mail address of the sender. Default: control@m
Sender Friendly Name	The name or alias that appears on the e-mail sent.
Reply-To Email	The e-mail address to which to send replies. If this field is left empty, the sender e-mail address is used.

WIN CM tab

Specify values for the parameters you want to change. If you quit without saving, settings are not saved. If you save, but do not quit, settings are saved anyway.

Table 15 lists the parameters displayed in the WIN CM tab and their descriptions.

Table 15 WIN CM tab parameters (part 1 of 2)

Parameter	Description
Logon Domain	The domain is determined by the value of this parameter if <domain> is not specified in <domain>\<username> in the **owner** parameter of the job definition. If the domain is not specified in the owner parameter or this parameter, the user profile is searched in the trusted domains.
	Note: BMC Software recommends that you do ***not*** specify a value for Logon Domain.
Domain Controller	Name of server managing access to resources and the database. Specify the name of the server in the field box. Default: **Blank**
Default Printer	Default printer for job SYSOUT files. Type a printer name in the field box or select a name from the list box. Default: **Blank**
Sysout Name	Determines the prefix for the Sysout file name. Valid values: ■ **MEMNAME** – the Sysout file prefix is the MEMNAME of the job. Default. ■ **JOBNAME** – the Sysout file prefix is the JOBNAME of the job.

Table 15 WIN CM tab parameters (part 2 of 2)

Parameter	Description
Add Job Object Statistics to Sysout	Flag that indicates how to manage job object processing statistics. ■ **Selected** – Statistics are added to the end of the SYSOUT file. Default. ■ **Not selected** – Statistics are not added to the SYSOUT file.
Logon As User	Flag that specifies which user account is used for the services to log on to. ■ **Selected** – Jobs are submitted with the permissions and environment variables of the specified user. ■ **Not selected** – Jobs are submitted with the permissions and environment variables of the local system account. Default. See "Determining the Owner of Jobs Run on the Agent" on page 27.
Job children inside job object	Flag that specifies if procedures invoked by a job can be run outside the Job Object. If so, this prevents a situation in which the original job remains in **executing** mode until the invoked procedure completes. ■ **N** – All procedures invoked by the job are run outside the job object. ■ **Y** – All procedures invoked by the job are run inside the job object. Default.
CD to user 'Home Dir'	Indication if CONTROL-M / Agent should change the directory to the user home directory as defined in the user profile before the user job runs. Valid values: ■ **Y** – The directory is changed. ■ **N** – The directory is not changed.
Run user 'Logon Script'	Indication if a user-defined logon script should be run by the CONTROL-M / Agent before running the standard user logon script. Valid values: ■ **Y** – The user-defined logon script is run, if it exists. ■ **N** – The user-defined logon script is not run.
CJK encoding	Determines the CJK encoding used by CONTROL-M / Agent to run jobs. For more information, see the *CONTROL-M Language and Customization guide*.

Click **SMTP Parameters** to display the SMTP Parameters tab. For a list of the parameters displayed in the SMTP Parameters tab and their descriptions, "SMTP Parameters tab" on page 58.

Additional Control Module tabs

In the Agent Configuration utility, you can manage all of the applications using the relevant CM's tab.

When you install a CM, a *cm_name*.xml data file is placed in the **ctm/data/GUI/** directory on UNIX computers or the **DATA\GUI** directory on Windows computers. The Agent Configuration utility reads each data file and creates a tab for the CM with each of the CM management applications.

NOTE

CM versions released prior to version 6.2.01 may not have the XML file and the respective CM tab.

For information about each of the CM management applications, see the relevant CM documentation.

Command_line utilities

This section describes the agent configuration command_line utilities. If the user running the utility is not an administrator, changes made to agent configuration parameters will not be saved. This is only relevant for the ctmagcfg and ctmwincfg utilities.

ctmagcfg

NOTE

This utility can also be accessed as a Java application. For more information, see the "Agent configuration utility" on page 53.

To access the ctmagcfg utility, enter **ctmagcfg** from a command prompt.

For an explanation of the parameters in the ctmagcfg utility, see Table 10 and Table 11.

ctmunixcfg

NOTE

This utility can also be accessed as a Java application. For more information, see the "OS CM tab (UNIX)" on page 57.

To access the ctmunixcfg utility, enter **ctmunixcfg** from a command prompt.

For a description of the parameters in the ctmunixcfg utility, see Appendix A, "Configuration parameters". If the parameter you want to modify is not listed, see the "Agent configuration utility" on page 53.

ctmwincfg

NOTE

This utility can also be accessed as a Java application. For more information, see the "WIN CM tab" on page 58.

To access the ctmwincfg utility, enter **ctmwincfg** from a command prompt.

For a description of the parameters in the ctmwincfg utility, see Appendix A, "Configuration parameters". If the parameter you want to modify is not listed, see the "Agent configuration utility" on page 53.

ctmfw utility (File Watcher)

The **ctmfw** utility can be used to detect

- successful completion of a file transfer activity
- creation of a file
- deletion of a file

ctmfw can be used before activating a job or before performing a task (for example, sending a shout message or adding / deleting conditions) that is dependent upon creation or deletion of a file.

The ctmfw utility runs as a process on a client computer. The process waits for the creation or deletion of specified files.

- For a file transfer activity, when the file is detected, the job continues to monitor the size of the file. When the file reaches a specified minimum size and does not increase in size for a specified period of time, the File Watcher utility either completes with a status of **OK** or executes a specified **DO** action. **DO** actions can consist of adding or deleting conditions or executing a command.

- For file creation, file size is ignored if a wildcard is specified as part of the filename unless the mon_size_wildcard parameter is set to **Y**.

- For file deletion, ctmfw must first detect the existence of the file before it can detect its deletion.

The ctmfw utility can also be run from the command_line, or be invoked to detect either a single file or multiple files.

Usage as a service

As a service, **ctmfw** takes its parameters (rules) during startup from the **rull.dat** file whose full path name is specified in **<CONTROL-M/Agent>\data\ctmfw.cfg**.

To change one or more rules, change the contents of the **rull.dat** file or specify the full path name of a different file.

NOTE

The rull.dat file provided with CONTROL-M / Agent is a sample file and should be changed to reflect your requirements.

The full path name to the **ctmfw.cfg** configuration file must be specified under the following Microsoft Windows registry key that is generated automatically by the installation script:

```
HKEY_LOCAL_MACHINE\SOFTWARE\BMC Software\
CONTROL-M/FileWatcher\SYSPRM\File Watcher
Configuration File
```

The default value for this key is

```
<CONTROL-M/Agent_install_directory>\DATA\ctmfw.cfg
```

NOTE

BMC Software recommends that this default value not be changed.

The configuration file must contain the following line:
`-input <ruleFileName>`

The variable <*ruleFileName*> is the full path name of a rule file containing the File Watcher rules. The following is a sample rule file.

Figure 3 Sample Rull.dat file

```
INTERVAL 5
FROM_TIME 0001
MIN_SIZE 50
MIN_DETECT 5
WAIT_TIME 2
ON_FILEWATCH NONEXIST CREATE 10 3 1
THEN
DO_CMD "BAD FILE WAS CREATED IN 1 MINUTE"
DO_COND ON_2 0101 +
ELSE
DO_CMD "GOOD FILE WAS NOT CREATED IN 1 MINUTE"
DO_CMD ctmshout -USER ECS -MESSAGE "Running the Filewatcher on  with
default configuration!"
END_ON
```

Network Resources

The FileWatcher service running under the local system account cannot detect network resources (files located on remote systems). If you want the File Watcher to detect network resources, configure the FileWatcher Service to run under a regular user account.

FileWatcher Service Trace

When running as a service, **ctmfw** generates an execution log file. This file is saved in the CONTROL-M / Agent **proclog** directory under the following name:

```
U_CTMFW_<process_id>.log
```

By default, logs in the **proclog** directory are retained for 3 days. If the “maximum days to retain SYSOUT” parameter is set to a number higher than 3, logs are retained for the number of days specified in that parameter.

Sample trace file

Figure 4 Sample Trace File Output

```
2002/03/10 13:04:24 182 FW:set INTERVAL=3
2002/03/10 13:04:24 182 FW:set MIN_SIZE=4
2002/03/10 13:04:24 182 FW:set FROM_TIME=0909
2002/03/10 13:04:24 182 FW:ctmfw:command line 'ON_FILEWATCH tst CREATE'(arg#=3).
id=1.
2002/03/10 13:04:24 182 FW:ctmfw:command line 'ON_FILEWATCH prd CREATE 0 0 1 NOW
10'(arg#=8)id=2
2002/03/10 13:04:24 182 FW:ctmfw:command line 'ON_FILEWATCH abc DELETE 0 0 1 NOW
10'(arg#=8)id=3
2002/03/10 13:04:24 182 FW:File 'test' exists, its current size is 265.
id=1.
2002/03/10 13:04:24 182 FW:File 'test' has reached  the minimum size of 4.
size=265 bytes id=1.
2002/03/10 13:04:24 182 FW:File 'abc' does not exist. id=3.
2002/03/10 13:04:36 182 FW:File transfer was completed. The size of file 'test'
is 265. id=1.
2002/03/10 13:04:36 182 FW:Executing:<ctmcontb add    'aaa' '0101'>
2002/03/10 13:05:09 182 FW:Executing:< dir >
2002/03/10 13:05:27 182 FW:File 'prd' was not CREATED within the time limit.
id=2.
2002/03/10 13:05:27 182 FW:File prd will be scanned at 1315. id=2.
2002/03/10 13:05:27 182 FW:File 'abc' was not DELETED within the time limit.
id=3.
2002/03/10 13:05:27 182 FW:File abc will be scanned at 1315. id=3.
2002/03/10 13:05:30 182 FW:File prd, is out of time window. next time:1315, id=2.
2002/03/10 13:05:30 182 FW:File abc, is out of time window. next time:1315, id=3.
2002/03/10 13:15:01 182 FW:File prd, entered the time window from '1315' for
monitoring, id=2.
2002/03/10 13:15:01 182 FW:File abc, entered the time window from '1315' for
monitoring, id=3.
```

Usage as a utility

When running as a utility, ctmfw is invoked from the command line. Rules can be provided on the command line or by a rule file.

To watch a single file

The syntax of the ctmfw utility is:

```
ctmfw
      < mode (CREATE|DELETE)>  Default: CREATE
      < minimum detected size <number>
       [' '|Bytes(B)|Kilo(K)|Mega(M)|Giga(G)] >Default:0
      < interval between file search (seconds) > Default: 60sec
       < interval between filesize comparison iterations (seconds) >
       Default: 10sec
       < number of iterations while the size is static > Default: 3 iterations
```

```
        < time limit for the process (minutes). Default: 0 (no time limit)
         Effective while the file does not exists or,
         the file size is static and the minimum size
         was not reached >
        < monitor file size when wildcard is used > Default: N
         < starting  time for detecting files (HHMM or YYYYMMDDHHMM >
         Default: NOW
        < absolute stop time (HHMM or YYYYMMDDHHMM > Default: 0 ( No stop time )
        < minimal age of file ( modified time )
          format:xxxxYxxxxMxxxxDxxxxHxxxxMin > Default: 0
```

The parameters of the ctmfw utility are described in Table 16.

All parameters must be assigned a value, even if that value is zero. If only six values are specified, the default value for **mon_size_wildcard** is used. If five parameters are specified, default values for **wait_time** and **mon_size_wildcard** are used, and so forth.

EXAMPLE

ctmfw /home/watchedfile.txt CREATE 100 10

is resolved using default values for **mon_int**, **min_detect**, **wait_time**, and **mon_size_wildcard** as follows:

ctmfw /home/samplefile.txt CREATE 100 10 10 3 0 N

Table 16 ctmfw parameters (part 1 of 3)

Parameter	Description
FILE	Path of the file to be detected. The file name can include mask character ***** to represent any number of characters (including no characters) or **?** to represent any one character. **Note**: The path and file name must not exceed 214 characters.

Table 16 ctmfw parameters (part 2 of 3)

Parameter	Description	
mode	CREATE	Detects creation of a file. Default. File size is ignored if the filename parameter contains wildcards (unless the **monitor file size when wildcard is used** parameter is set to **Y**).
		Note: If a mask is specified for the filename, and the **monitor file size when wildcard is used** parameter is set to ■ **N**, the ctmfw utility will end **OK** after detection of the first file that matches the specified mask. ■ **Y**, the ctmfw utility will end **OK** after detection of the first file that matches the filename and file size.
		For more information about **monitor file size when wildcard is used**, see below.
	DELETE	Detects deletion of a file. When the ctmfw utility is run in this mode, it first checks for files that match the specified name. After a specified file is detected, the ctmfw utility checks at the specified interval for deletion of that file.
		Note: If a mask is specified as the filename, the ctmfw utility will end successfully only after all detected files that match the specified mask have been deleted.
minimum detected size	Minimum file size in bytes. This parameter is ignored if the **FILE** parameter contains wildcards (unless the **monitor file size when wildcard is used** parameter is set to **Y**) or if the **mode** parameter is set to **DELETE**. Default: **0** (any size detected).	
interval between file searches	Interval between successive attempts to detect the existence / deletion of a file (in seconds). Default: **60**	
interval between filesizecomparison iterations	Interval between attempts to monitor the size of a file after it is detected (in seconds). This parameter is ignored when using wildcards in **FILE** or when using **DELETE** mode. Default: **10**	
number of iterations while size is static	Number of attempts to monitor file size where the size remains static and greater than or equal to **minimum detected size** (indicating successful creation of the file). This parameter is ignored when using wildcards in **FILE** or when using **DELETE** mode. Default: **3**	
time limit for the process	Maximum time (in minutes) to run the process without detecting the file at its minimum size (**CREATE**) or detecting its deletion (**DELETE**). If the file is not detected / deleted in this specified time frame, the process terminates with an error return code, as described in Table 19. Default: **0** (no time limit).	
monitor file size when wildcard is used	Indicates whether file size should be monitored if the filename contains wildcards. This parameter is ignored if the filename does not contain a wildcard. Valid values: **N** – do not monitor file size. Default. **Y** – monitor the file size. If this parameter is set to **Y** and more than one file matches the specified mask, the ctmfw utility randomly selects one matching file, monitors its file size, and ignores all other matching files.	

Table 16 ctmfw parameters (part 3 of 3)

Parameter	Description
starting time for detecting files	Indicates an absolute time at which the utility starts monitoring the file. For example, 200512061400, means that at 2 PM on December 6th, 2005 the FileWatcher utility will stop watching the file. Alternatively, you can use the HHMM format, in which case the current date is used.
absolute stop time	Indicates an absolute time at which the file is no longer watched. For example, 200502061400, would mean that at 2 PM on February 6th, 2005 the FileWatcher utility will stop watching the file. Alternatively, you can use the HHMM format, in which case the current date is used.
maximal age of file	Indicates the maximum amount of time that can pass since the file you want to watch was last modified. For example, 2y3d5h means that after 2years, 3 days, and 5 hours has passed, the file will no longer be watched. Entering a value of 2H10Min, means that after 2 hours and 10 minutes has passed, the file will no longer be detected. This parameter is ignored if the mode parameter is set to **DELETE**. Default: **0**
minimal age of file	Indicates the minimum amount of time that must have passed since the file you want to watch was last modified. For example, 2y3d5h means that 2years, 3 days, and 5 hours must pass before the file will be watched. Entering a value of 2H10Min, means that 2 hours and 10 minutes must pass before the file will be detected. This parameter is ignored if the mode parameter is set to **DELETE**. Default: **0**

Importing the File Watcher panel (optional)

The following procedure ensures that File Watcher job parameters (in the File Watcher panel) are displayed in the Job Editing form in CONTROL-M/EM and CONTROL-M/Desktop.

Before you begin

Ensure that the CONTROL-M/Agent application CD is mounted or easily accessible through your network.

To import the File Watcher panel

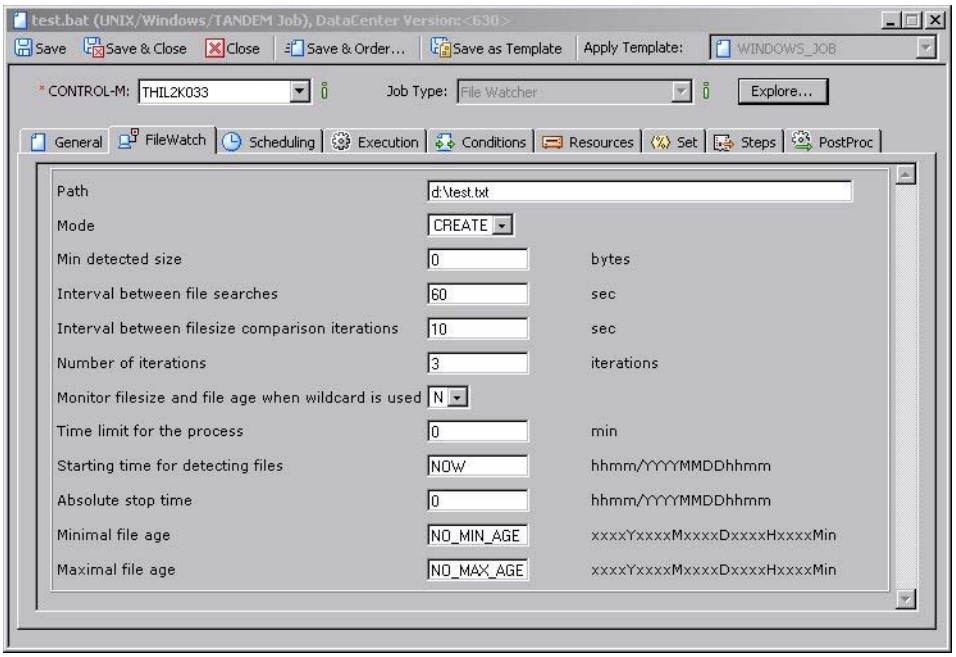
Perform this procedure on every computer that has CONTROL-M/Enterprise Manager client component(s) installed.

1 In CONTROL-M/Desktop, choose **Tools => Import Applications Forms**. A browse window is displayed.

2 Navigate to the **Forms** directory on the installation CD. Select the **FileWatch.xml** file and click **Import**.

3 Shut down the CONTROL-M/Enterprise Manager GUI, and then restart it. This enables the newly imported File Watcher panel functionality.

Figure 5 FileWatch - File Watcher panel



For more information about the parameters in the File Watcher panel, see Table 16 on page 65. For more information about the Job Editing form, see the *CONTROL-M/Enterprise Manager User Guide*.

NOTE

The path and file name must not exceed 214 characters

To watch multiple files

Use the following command to invoke the ctmfw utility for multiple files:

ctmfw -input <*ruleFileName*>

The variable <*ruleFileName*> is the complete path name of the file containing the definitions for each file to be detected.

Sample rule file

Figure 6 displays a sample rule file. In this sample:

- **#** indicates comments.
- Default values are shown for all global parameters.
- **<action>** refers to any of the actions described in Table 18.

Figure 6 Sample rule file

```
#*****************************************************************
# Global Parameters
INTERVAL <60>  # Sleep interval (seconds)
MIN_SIZE 4Kilo
MIN_AGE  3M24D4h5min
FROM_TIME <0000>  # Starting time for all files (hhmm)
MIN_SIZE <0>  # Minimum size for all files (bytes)
MIN_DETECT <3>  # Number of iterations for all files
WAIT_TIME <0>  # Time limit for all files (minutes)

# ON_FILEWATCH statements
ON_FILEWATCH <filename>(absolute path) [CREATE/DELETE] [min_size] [min_detect]
[wait_time]
            [start_time] [cyclic_interval] [wildcards] [minimal_file_age]
THEN
<action>
ELSE
<action>
END_ON
#*****************************************************************
```

If a wildcard is used in the file name, the found file can be referenced as %FILENAME%.

EXAMPLE

```
INTERVAL 10
ON_FILEWATCH /controlm/datafile*.txt CREATE
THEN
DO_COND %FILENAME% 0101 +
DO_CMD move %FILENAME% /ctm/workfile.txt
ELSE
DO_COND %FILENAME% 0101 -
END_ON
```

NOTE

All global parameters must be delimited by the new line character.

The Rules file contains two sections:

- Global parameters, whose default values apply to all the files in the rule file.
- **ON_FILEWATCH** statements identifying which files to detect, specific criteria for each file, and the action to take upon detection or non-detection. Any number of **ON_FILEWATCH** statements can appear in a Rules file.

NOTE

All keywords must be entered in uppercase.

Table 17 Rule file global parameters (part 1 of 2)

Param	Description
INTERVAL	Sleep interval (in seconds) between successive scans for all the files. This parameter replaces individual **sleep_int** and **mon_int** parameters for each file. Default: **10**
MIN_SIZE	Minimum file size in bytes. This parameter is ignored if the **FILE** parameter contains wildcards (unless the **monitor file size when wildcard is used** parameter is set to **Y**) or if the mode parameter is set to **DELETE**. Default: **0** (any size detected).
MIN_AGE	Indicates the minimum amount of time that must have passed since the file you want to watch was last modified. For example, 2y3d5h means that 2years, 3 days, and 5 hours must pass before the file will be watched. This parameter is ignored if the mode parameter is set to **DELETE**. Default: **0**
MAX_AGE	Indicates the maximum amount of time that can pass since the file you want to watch was last modified. ■ If MAX_AGE = 0, any change to the file timestamp means that the condition is met. ■ IF MAX_AGE = 10 Min and if the amount of time of the watched file that has passed is less than 10 minutes, then the condition is met. This parameter is ignored if the mode parameter is set to **DELETE**. Default: **0**
FROM_TIME	Starting time for detecting all the files (default FROM_TIME). Used with WAIT_TIME to identify the time frame for detecting and monitoring the files. This parameter is expressed in 24-hour, hhmm format. Default: **0000** or **Now**
STOP_TIME	Indicates an absolute time at which the file is no longer watched. For example, 200502061400, means that at 2 PM on February 6th, 2005 the FileWatcher utility will stop watching the file. You can also use the HHMM format, which uses the current date, plus the HHMM entered. Default: **0** (meaning, no stop time) **Note**: STOP_TIME can only be used as a global parameter.
CYCLIC_INTERVAL	Indicates the interval between multiple operations of detecting the file (in minutes). This interval must be greater than the value for WAIT_TIME. If the cyclic_interval is 0, only one attempt to detect the file will be performed. Default: **0**

Table 17 Rule file global parameters (part 2 of 2)

Param	Description
MON_SIZE_ WILDCARD	Indicates whether file size should be monitored if the filename contains wildcards. This parameter is ignored if the filename does not contain a wildcard. Valid values: **N** – do not monitor file size. Default. **Y** – monitor the file size. If this parameter is set to **Y** and more than one file matches the specified mask, the ctmfw utility randomly selects one matching file, monitors its file size, and ignores all other matching files.
WAIT_TIME	Maximum time (in minutes) to run the process without detecting the file at its minimum size (**CREATE**) or detecting its deletion (**DELETE**). If the file is not detected / deleted in this specified time frame, the process terminates with an error return code, as described in Table 19. Default: **0** (no time limit).

NOTE

For a description of the ON_FILEWATCH parameters, see Table 16 on page 65.

If any mandatory parameter is omitted from a Rules file, the default value for that parameter is used. Parameters entered for **ON_FILEWATCH** statements override the default values. If entered, they must appear in the order shown in Figure 6.

Table 18 ctmfw – valid actions

Action	Description
DO_COND <condition name> <condition date> <+|->	Add (+) or delete (-) a condition.
DO_CMD <command>	Execute a valid command under the command interpreter. Full path names are required for files.
DO_OK	Terminate an ON_FILEWATCH statement with status **OK**. If there is more than one file in the Rule file, the result displayed is that of an AND algorithm.
DO_NOTOK [exit code]	Terminate an ON_FILEWATCH statement with status **NOTOK**. Exit code is optional and replaces the standard return code, as described in Table 19.
DO_EXIT [exit code]	Terminate ctmfw with the user-defined exit code.

- If the file is detected and the size remains static within the time frame (**CREATE**) or the file has been deleted (**DELETE**), the **DO** commands in the **THEN** block are executed.
- If the file is not detected or deleted within the time frame, the statements following the **ELSE** block are executed.

- ctmfw terminates when all the files in the Rules file have been processed.

NOTE

If an ON_FILEWATCH statement contains a **cyclic_interval** parameter, ctmfw will only stop monitoring a file on a DO_OK or DO_NOTOK action.

Example 1

The ctmfw utility is invoked to watch multiple conditions. The definitions the ctmfw utility uses for watching each file are contained in a rule file.

The following instructions are defined in the Rules file:

- The sleep interval between succeeding scans must be 10 seconds.
- If the ctmfw utility detects that the datafile.txt file in the **/home/controlm** directory is created in the specified time interval, then:
  - the **datafile** condition dated 1 January must be added.
  - The command interpreter must execute the command to move the contents of the file /home/ctm/datafile.txt to /home/ctm/workfile.txt.
- If the ctmfw utility detects that the **datafile.txt** file in the **/home/controlm** directory is not created in the specified time interval, then condition **datafile** dated 1 January must be deleted.
- When the ctmfw utility detects that the **/home/ctm/tempfile.txt** file is deleted, condition **tempfile** dated 1 January must be deleted.

```
#*****************************************************************
INTERVAL 10
ON_FILEWATCH \ctmsrv\datafile.txt CREATE
THEN
DO_COND datafile 0101 +
DO_CMD move \ctm\datafile.txt \ctm\workfile.txt
ELSE
DO_COND datafile 0101 -
END_ON
ON_FILEWATCH \ctm\tempfile.txt DELETE
THEN
DO_COND tempfile 0101 -
END_ON
#*****************************************************************
```

Example 2

A job processing definition is created to implement a File Watcher job. The file must arrive between 19:00 and 22:00, and be created in the **/tmp** directory under the name **trans.dat**. The minimum file size is 100 bytes. The detection process should be performed each minute. The file size monitored every 10 seconds, and the number of intervals where the file size remains static is 5. If the file is not detected by 22:00, an alert should be sent to CONTROL-M/Enterprise Manager.

Parameter	Value
Job Name	FileWatch
Mem Name	FileWatch
Owner	<control_m_user>
From Time	1900
Command line	ctmfw "\tmp\trans.dat" CREATE 100 60 10 5 180
On Statement/Code processing:	
Stmt	*
Code	COMPSTAT=0
Do Cond	file_trans_dat_ok **Date:** ODAT **Sign:** +
Stmt	*
Code	COMPSTAT=1
Do Shout	**To:** CONTROL-M/Enterprise Manager **Text:** "File trans.dat did not arrive on time"

Return codes

The return codes listed in Table 19 are issued by the ctmfw utility after detecting if a file is created or deleted in the specified time frame.

Table 19 ctmfw – return codes

Return code	Description
0	File successfully created or deleted (file arrived in the specified time frame and file size is above or equal to the minimum specified size).
1	■ Utility failed. For example, because of a syntax error. ■ A DO_NOTOK statement occured, but no user-defined exit code was provided for the DO_NOTOK statement.
7	Indicates that the ctmfw request timed out. That is, the file was not detected in the specified time frame.

File Watcher silent mode registry key

The FileWatcher service does not open an additional window during execution. If you want visual feedback while running the service, the following registry key setting must be changed to **N**.

```
HKEY_LOCAL_MACHINE\SOFTWARE\BMC_Software\
CONTROL-M\FileWatcher\SYSPRM\Silent_Mode
```

shagent utility

The shagent utility (*UNIX only*) checks that the **p_ctmag** and **p_ctmat** processes are running. It can be invoked only from the CONTROL-M / Agent computer. The utility has no parameters.

If the Persistent Connection parameter is set to **Y**, the utility verifies that the p_ctmar process is running.

From the operating system prompt, specify the following command:

```
shagent
```

Sample Output

If the Router process is running, output similar to the following is displayed:

```
root         7660   0:00 p_ctmag
root         7644   0:00 p_ctmar
root         7745   0:29 p_ctmat
```

ctmpwd utility

The ctmpwd utility (*Windows only*) adds, updates, and deletes CONTROL-M / Agent users and passwords. In addition, it changes security settings for the agent directories and cmd.exe. It also lists all users in the CONTROL-M / Agent password file. (This utility replaces the ctmcpt utility in earlier versions.)

To run ctmpwd, you must be an administrator on the computer. In addition, Windows 2000 users user running the utility must have the proper privileges defined in the Act as part of operating system parameter of the Local Security Settings application on the target computer.

NOTE

You must manually give **Logon as a batch job** rights to a new user. (See "How to Assign User Rights to Agent Users" on page 28.)

Syntax

```
CTMPWD -ACTION ADD|UPDATE|DELETE|LIST [-USER <user name>] [[-OLD_PASSWORD
<value>] -PASSWORD <value>] [-ADMIN_PASSWORD <value>]] -AGENT <agent name> -GROUP
<group name>
```

Examples

add a user and password

```
ctmpwd -action add -user user1 -password 12345
```

add the administrator user

```
ctmpwd -action add -user admin -password abcde
```

update a password

```
ctmpwd -action update -user user1 -old_password 12345 -password 67890
or
ctmpwd -action update -user user1 -admin_password abcde -password
67890
```

delete a user

```
ctmpwd -action delete -user user1 -password 12345
or
ctmpwd -action delete -user user1 -admin_password abcde
```

list all users

```
ctmpwd -action list
```

add a user to agent Saturn

```
ctmpwd -action add -user user3 -password 654321 -agent Saturn
```

Table 20 ctmpwd utility – parameters

Parameter	Description
action	Function to be executed. Valid values: **add**, **update**, **delete**, and **list**.
user	Name of the user. **Note**: When adding users, the user name must not exceed 20 characters.
old_password	Current password for the **update** function.
password	Current password for the **delete** function. New password for the **add** and **update** functions.

Table 20 ctmpwd utility – parameters

Parameter	Description
admin_password	Password for the CONTROL-M / Agent administrator when executing the **update** or **delete** function if the old_password is not known.
verify	Verifies that the user and password. Valid values are: ■ **Y** (default) ■ **N** - Does not verify that the user and password are correct.
group	Adds the group SID (instead of the user SID) to cmd.exe and the agent directories.
agent	Name of the agent that the utility is designated to run on. For more information, see "Invoking a utility on a computer with Multiple Agents (Windows)" on page 50.

- In the following example, the ctmpwd utility enables the CONTROL-M / Agent administrator to modify passwords for users who have forgotten their password.

EXAMPLE

```
-admin_password
```

BMC Software recommends that the administrator first use the following command to establish a password for user ADMIN:

```
ctmpwd -action add -user ADMIN -password <user_admin_password>
```

- In the following example, the user is added but the group's SID is registered.

EXAMPLE

```
ctmpwd -action add -user user1 -password user1 -group Everyone
```

Agent check utility

The CONTROL-M Agent Check Utility (ACU) is a tool that collects information and diagnostic data about the CONTROL-M Agent installation, execution state, and target environment. The data collected by the ACU is designed to assist CONTROL-M / Agent administrators and BMC Software technical support engineers to troubleshoot, fine-tune, and maintain the CONTROL-M / Agent.

With this tool, you can send generated reports to interested parties using e-mail or FTP to BMC Software (ftp://ftp.bmc.com/incoming). You can print the report to a hierarchical XML file, or save the report as a text file. In addition, you can set the agent debug level and download the most recent agent and CM patches.

This utility is located at **/<*agent_directory*>/ctm/exe/** and can be run from either the command line or as a Java application.

To start the ACU application

1 Log on to the target CONTROL-M/Agent under the agent account on which ACU is installed.

2 Set the display variable using the command

```
setenv DISPLAY <host name>:0.0
```

3 Enter the `acu_gui` command to run the Java application

-or-

enter `acu` to run the application from the command line.

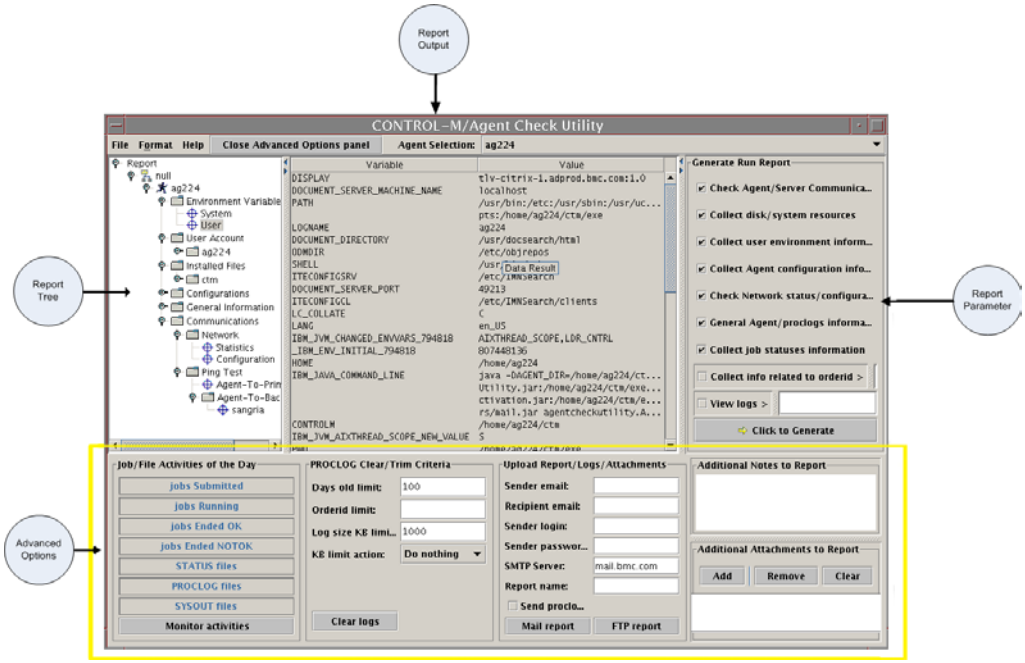
For more information about running the utility from the command line, see "Command line usage" on page 80.

Using ACU

The ACU is divided into the following panels:

- Report tree
- Report output
- Report parameters selection
- Advanced Options panel

Figure 7 Agent Check Utility window



Report Parameter selection

The right-most panel of the ACU is used to select the data that you want to include in your report.

Table 21 Report parameters

Parameter Name	Description
Check Agent/Server Communication	Selecting this option runs the ag_ping utility. This utility verifies that CONTROL-M/Server is active on the Server computer connected to the agent computer.
Check Network status/configuration	Selecting this option provides information about the network configuration and statistics using the netstat and ipconfig commands.
Collect Agent configuration information	Selecting this option provides information about the agent on which you ran the report. This also checks the dll versions and configurations for the agent and CMs installed, and lists the files installed.
Collect disk/system resources	Selecting this option provides system information about the memory in use, free disk space, number of processes running, and so on.
Collect info related to orderid	Selecting this option collects information from files with names containing the specified orderid or information related to the specified orderid.

Table 21 Report parameters

Parameter Name	Description
Collect job statuses information	Selecting this option provides you with information about the number of jobs processed since the last NewDay procedure, their status, job start time, job end time, and so on.
Collect user environment information	Selecting this option provides Java-applications system information and user specific information, such as root directories, domain, path, and so on.
General Agent/proclogs information	Selecting this option enables you to activate the newly configurated debug level and start collecting the log files from the PROCLOG directory. The logs can then be transferred using e-mail or ftp (if it is a large size zip file) in zipped format.
View logs	Selecting this option opens a browser window from which you can select log files to attach to the generated report. You can then view the content of the attached log files in the output panel.

Report tree

Once you select the areas for which you want to collect data and generate the report, a report tree appears on the left side of the screen. This tree lets you drill down into the various report parameters and select them for viewing in the Output panel.

Output panel

The Output panel is where the data of the selected parameters is displayed. You can save, print, or email the information for further analysis.

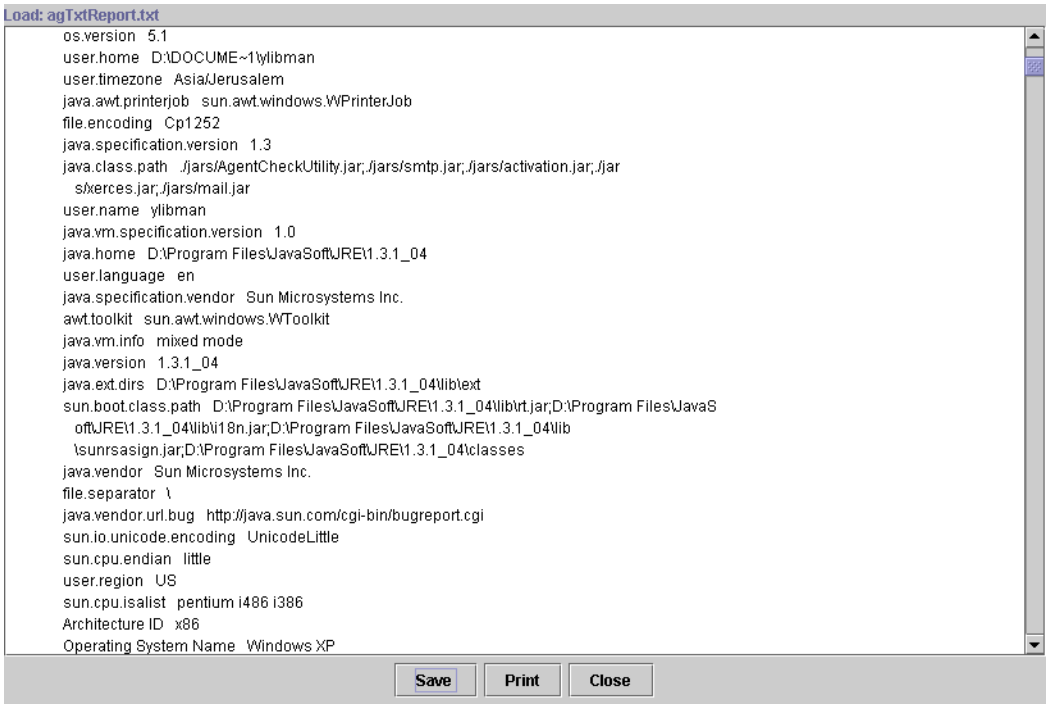
Advanced Options

The Advanced options panel enables you to view additional information about the day's activities, define parameters by which the PROCLOG files are saved, enter mail and FTP information for sending and uploading reports, and so on.

Printing

You can print a generated or uploaded report or save it as a text file using the **File > Print** menu option.

Figure 8 Print window



To print the report, click **Print**. To save to a text file, click **Save** and enter a file name in the Save window.

Command line usage

To run ACU as a command_line utility, enter the command **acu** with the relevant parameters. When ACU runs as a command_line utility, a report of the ACU output is automatically saved to the **/<*agent_directory*>/temp** directory.

Syntax

```
acu   [agent/<name>] [all] [silent] [system] [environment]
        [configuration] [ping] [analyze] [jobinfo] [network]
        [orderid/<orderid num>]
        [ftplogs/<ftp_server> ftpdir/<ftpdir> filename/<filename>]
        [mailreport/<from>/<to>/<id>/<pwd> mailsmtp/<smtp server>
[maillog]]
```

Table 22 lists the command_line parameters and their descriptions.

Table 22 Command_line parameters

Parameter	Description
agent <name>	Used to specify an agent when more than one agent installation exists. Optional. When used, this parameter must be first.
all	Collects information for all options, including ping, environment, system, network, configuration, and analyze.
analyze	Analyzes the agent logs and provides information about the possible symptom and its solution.
configuration	information about the agent on which you ran the report.
environment	Collects system information and user-specific information, such as root directories, domain, path, and so on.
filename	The name of the ZIP file you are uploading to the BMC Software ftp site.
ftpdir	Specify the directory into which the ZIP file is uploaded.
ftplogs	Zip the agent PROCLOG directory and upload it to the BMC ftp server.
jobinfo	Provides information about the number of jobs processed since the last NewDay procedure, their status, job start time, job end time, and so on.
maillog	Zip the agent PROCLOG directory and attach it to the e-mail.
mailreport	Output the report to a specific e-mail address.
mailsmtp	Specify the smtp server of the corresponding e-mail address.
network	Collects information about the network configuration and statistics.
orderid	Collects information from the files with names containing the specified orderid or information related to the specified orderid.
ping	Checks the connection between CONTROL-M/Agent and CONTROL-M/Server.
silent	Run the utility with no on-screen output.
system	Collects information about system resources.

Command line examples

- In the following example, ACU collects all the information about a specific agent and outputs it to your screen and to the **/<*agent_directory*>/temp** folder.

EXAMPLE

```
>acu all
```

- In the following example, ACU checks the connection between the agent and server and collects information about the system, environment, and agent configurations.

EXAMPLE

```
>acu ping network configuration system environment
```

- In the following example, ACU collects all the information about a specific agent and sends the information, including all the logs from the **/<*agent_directory*>/proclog** directory, to a specific e-mail address using the corresponding smtp server.

EXAMPLE

```
>acu all mailreport/user@domain.com/support@bmc.com
mailsmtp/mail.domain.com maillog
```

- In the following example, ACU collects all the information about a specific agent and uploads the information to the BMC ftp site.

EXAMPLE

```
>acu all ftplogs/ftp.bmc.com ftpdir/incoming filename/case_4684
```

Configuration parameters

This appendix describes configuration parameters, the files were they are stored, and the utilities that can be used to modify them.

The following topics are discussed in this appendix:

CONFIG.dat file configuration parameters

The CONTROL-M / Agent configuration parameters in Table 23 are stored in the **CONFIG.dat** file. These parameters can be modified using the "Agent configuration utility" on page 53. Some of these parameters are also described in the customization parameters chapter in the *CONTROL-M/Server for UNIX and Microsoft Windows Administrator Guide*.

Table 23 Agent – CONFIG parameters (part 1 of 4)

Parameter	Description
AGENT_STATUS (UNIX only)	This parameter is read by the **rc.agent_user** startup script to determine whether the agent should be started. Valid values are: ■ Started ■ Stopped
AG_LOG_ON	Indicates if the **ctmag_<year><month><day>.log** file is generated (**Y**) or not (**N**). Default: **Y**

Table 23 Agent – CONFIG parameters (part 2 of 4)

Parameter	Description
AGCMNDATA	<Port number> / <Timeout> for the Server-to-Agent port. Port number specifies agent computer port that receives data from the Server computer. Verify that this port is not used for any other purpose. Must match Server-to-Agent port number in CONTROL-M / Server. Must be between **1024** and **65533** inclusive. Default: **7006**. The timeout value is the **COMTIMOUT** communication job-tracking timeout in seconds. Mandatory. Example: **7006/30**
AGENT_DIR	Location of files used by CONTROL-M / Agent.
ALLOW_COMM_INIT	Determines if the agent can open a connection to the server when working in persistent connection mode. Valid values Y, N. Default: **Y**
AR_AG_COMM_PORT	Internal port used only when the agent is working in persistent connection mode. This port is selected by the installation, which validates that it is free. If this port is not free when the agent is started in persistent connection mode, or when it is shifted from transient to persistent during runtime, the agent will automatically find a new free port and update the parameter accordingly.
AR_AT_COMM_PORT	Internal port used only when the agent is working in persistent connection mode. This port is selected by the installation, which validates that it is free. If this port is not free when the agent is started in persistent connection mode, or when it is shifted from transient to persistent during runtime, the agent will automatically find a new free port and update the parameter accordingly.
AR_UT_COMM_PORT	Internal port used only when the agent is working in persistent connection mode. This port is selected by the installation, which validates that it is free. If this port is not free when the agent is started in persistent connection mode, or when it is shifted from transient to persistent during runtime, the agent will automatically find a new free port and update the parameter accordingly.
ATCMNDATA	<Port number> / <Timeout> for the Agent-to-Server port. Port number specifies the Server computer port that receives data from the agent computer. Verify that this port is not used for any other purpose. This value must match the Agent-to-Server Port Number in CONTROL-M / Server. The value must be a number between **1024** and **65533** inclusive. Default: **7005**. The Timeout value is the **COMTIMOUT** communication job-tracking timeout in seconds. Mandatory. Example: **7005/30**. **Note:** Increasing the Timeout value reduces agent performance.
AUTOEDIT_INLINE	Flag that indicates whether all AutoEdit variables will be set as environment variables in the script. Valid values: **Y** (yes), **N** (no). Default: **Y**
CM_APPL_TYPE	Default control module. Default: **OS**

Table 23 Agent – CONFIG parameters (part 3 of 4)

Parameter	Description
CMLIST	List of Control Modules. For internal use only.
COMM_TRACE	Flag indicating whether communication packets that CONTROL-M/Agent sends to and receives from CONTROL-M/Server are written to a file. Valid values: **1** (on), **0** (off). Default: **0** (off). If set to **1**, separate files are created for each session (job, ping, and so forth). This parameter can only be changed after completing the installation.
COMMOPT	Determines CONTROL-M communication is secured using the SSL protocol. Valid values SSL=Y, SSL=N. Default SSL=N
COMMRETSLP	Time in seconds (integer value} to wait between each attempt to attach to the CONTROL-M/Server. Default: **1**
CTMPERMHOSTS	<one or more TCP/IP addresses or DNS names separated by |>. Each value identifies an authorized CONTROL-M/Server host where a backup CONTROL-M/Server is installed. (This parameter was previously called Mirror CONTROL-M/Server Host Name.) Specify this parameter if one or more CONTROL-M/Servers have been designated as backups and can connect to this agent in case of failover. For information about backup server configuration, see the *CONTROL-M/Server for UNIX and Microsoft Windows Administrator Guide*. Mandatory. At least one primary host name should be specified. Example: **192.138.28.121 | aristo.isr.bmc.com/mybksys1 | 192.138.28.123**
CTMS_ADDR_MODE	**{IP}** If this parameter is set to **IP**, the IP address instead of the host name is saved in CTMS_HOSTNAME. Use this parameter when CONTROL-M runs on a computer with more than one network card.
CTMSHOST	CONTROL-M/Server host name. Name of the primary host running CONTROL-M/Server.
DBGLVL	CONTROL-M/Agent diagnostic level (for use by Technical Support). Determines types of diagnostic messages generated. This parameter is normally set to zero (no diagnostics). Range: **0 - 4**. Default: **0**
EVENT_TIMEOUT	Job Tracking Timeout. Tracker event timeout in seconds. Default: **120**
I18N	Determines CONTROL-M mode of support for foreign languages. Valid values Latin-1, CJK default: Latin-1
LISTEN_INTERFACE	The network interface the agent is listening on. By default it is set to *ANY, meaning that the agent is listening on all available interfaces. It can be set to a specific hostname or IP address so that the agent port is not opened in the other interfaces.
LOCALE	(*UNIX only*) Determines the locale under which the agent processes run. Valid only when the agent is working in Latin-1 mode. Supported locales are specified in.
LOCALHOST	This parameter is no longer relevant.

Table 23 Agent – CONFIG parameters (part 4 of 4)

Parameter	Description
LOGICAL_AGENT_ NAME	Logical name of the agent. The value specified should match the name the agent is defined by in CONTROL-M / Server. By default it is set to the agent host name. However, it can differ when either a cluster installation or the agent host name has aliases.
LOGKEEPDAYS	Number of days to retain agent proclog files. After this period, agent proclog files are deleted by the New Day procedure. Default: **1** **Note:** This parameter is relevant only if CONTROL-M / Server does not pass a parameter that determines how many days to keep log files.
PERSISTENT_ CONNECTION	Determines if the agent is working in persistent or transient communication mode. Valid values Y, N. Default: **N**
PROTOCOL_ VERSION	Server-Agent communication protocol version. Default: **07**
TRACKER_EVENT_ PORT	Number of the port for sending messages to the Tracker process when jobs end.
UTTIMEOUT	Maximum time (in seconds) the agent waits after sending a request to CONTROL-M / Server. This timeout interval should be longer than the TCP / IP Timeout. Recommended value and default: **120**

CM for UNIX configuration parameters

The Control Module configuration parameters in Table 24 are stored in the **OS.dat** file.Most of the parameters described in the table can be modified using the ctmunixcfg utility, described on page 61.

Table 24 CM for UNIX – OS.dat parameters (part 1 of 2)

Parameter	Description
APPLICATION _LOCALE	Determines the CJK encoding used by CONTROL-M / Agent to run jobs. For more information, see the *CONTROL-M Language and Customization guide*.
CM_TEMP_SCRIPTS _DIR	Default path for saving temporary scripts.
CTM_PARM_ENC	Character used to enclose job processing parameters passed to jobs by CONTROL-M / Agent. Any character or string can be specified. A blank space (in single or double quotes) is valid.
CTM_PRM_DONT _DELETE	By default, temporary scripts generated from jobs are deleted at the end of job execution. If this value is set to **YES**, temporary scripts are not deleted.

Table 24 CM for UNIX – OS.dat parameters (part 2 of 2)

Parameter	Description
CTM_PRM_KSH _FLAGS	Indicates the shell flag that will be used to run the job script. For more information, see "Specifying the shell type" on page 32. Valid values: **-x**, **-v, n**. Default: **-x**
CTM_PRM_SH _FLAGS	Indicates the shell flag that will be used to run the job script. For more information, see "Specifying the shell type" on page 32. Valid values: **-x**, **-v**, **n**. Default: **-x**
CTM_SU_PATH	Alternative path that CONTROL-M / Agent will use to look for the **su** command. Default: **/bin/su**
PRINTER_NAME	Default printer for job output (SYSOUT).
PROCLOG_MODE	Octal value indicating file access mode of the Proclog (output) file. **777** indicates the highest level of access.
RJX_DETAILS_TO _SYSOUT	Determines whether to include details related to the remote connection in the job sysout of jobs executed on a remote host. Valid values: **Y** or **N**. Default: **Y**
SMTP_PORT _NUMBER	The port number on which the SMTP server communicates. Default: 25
SMTP_REPLY_TO _ EMAIL	The e-mail address to which to send replies. If this field is left empty, the sender e-mail address is used.
SMTP_SENDER _EMAIL	The e-mail address of the sender. Default: control@m
SMTP_SENDER _FRIENDLY_NAME	The name or alias that appears on the e-mail sent.
SMTP_SERVER _NAME	The name of the SMTP server.
SYSOUT_MODE	Octal value indicating file access mode of the Sysout (output) file. **777** indicates the highest level of access.
SYSOUT_NAME	{JOBNAME | MEMNAME} If set to **JOBNAME**, parameter **Jobname** is used in the SYSOUT file instead of parameter **Memname**. Default: **MEMNAME**
TRANSLATE_$0	If set to **Y**, reserved variable **$0** specifies whether instances of $0 in the job script should be replaced before the script is run. If set to **N**, this functionality is disabled. For more information, see "Use of the $0 Reserved Variable" on page 37.

CM for Windows configuration parameters

Most of the Control Module configuration parameters described in Table 25 can be modified by using the ctmwincfg utility. For information about this utility, see "ctmwincfg" on page 61. These parameters are used by the Control Module for Windows. For information about these parameters, see "WIN CM tab" on page 58.

Specify values for the parameters you want to change. If you quit without saving, settings are not saved. If you save, but do not quit, settings are saved anyway.

Table 25 CM for Windows – OS parameters (part 1 of 3)

Parameter	Description
CHANGE_TO_USER_HOMEDIR	Indication if CONTROL-M / Agent should change the directory to the user home directory as defined in the user profile before the user job runs. Valid values: ■ **Y** – The directory is changed. ■ **N** – The directory is not changed.
DFTPRT	Default printer for job SYSOUT files. Type a printer name in the field box or select a name from the list box. Default: **Blank**
DOMAIN_SERVER	Name of server managing access to resources and the database. Specify the name of the server in the field box. Default: **Blank**
DOMAIN	The domain is determined by the value of this parameter if <domain> is not specified in <domain>\<username> in the **owner** parameter of the job definition. If the domain is not specified in the owner parameter or this parameter, the user profile is searched in the trusted domains.
	Note: BMC Software recommends that you do ***not*** specify a value for Logon Domain.
JO_STATISTIC	Flag that indicates how to manage job object processing statistics. ■ **Selected** – Statistics are added to the end of the SYSOUT file. Default. ■ **Not selected** – Statistics are not added to the SYSOUT file.
JOB_WAIT	Flag that specifies if procedures invoked by a job can be run outside the Job Object. If so, this prevents a situation in which the original job remains in **executing** mode until the invoked procedure completes. ■ **N** – All procedures invoked by the job are run outside the job object. ■ **Y** – All procedures invoked by the job are run inside the job object. Default.
LOGON_AS_USER	Flag that specifies which user account is used for the services to log on to. ■ **Selected** – Jobs are submitted with the permissions and environment variables of the specified user. ■ **Not selected** – Jobs are submitted with the permissions and environment variables of the local system account. Default. See "Determining the Owner of Jobs Run on the Agent" on page 27.

Table 25 CM for Windows – OS parameters (part 2 of 3)

Parameter	Description
RJX_DETAILS_TO_SYSOUT	Determines whether to include details related to the remote connection in the job sysout of jobs executed on a remote host. Valid values: **Y** or **N**. Default: **Y**
RUN_USER_LOGON_SCRIPT	Indication if a user-defined logon script should be run by the CONTROL-M / Agent before running the standard user logon script. Valid values: ■ **Y** – The user-defined logon script is run, if it exists. ■ **N** – The user-defined logon script is not run.
SMTP_PORT_NUMBER	The port number on which the SMTP server communicates. Default: 25
SMTP_SENDER_EMAIL	The e-mail address of the sender. Default: control@m
SYSOUT_NAME	Determines the prefix for the Sysout file name. Valid values: ■ **MEMNAME** – the Sysout file prefix is the MEMNAME of the job. ■ **JOBNAME** – the Sysout file prefix is the JOBNAME of the job.

Table 25 CM for Windows – OS parameters (part 3 of 3)

Parameter	Description
WRAP_PARAM _QUOTES	Indication of how parameter values (%%PARMn....%%PARMx) are managed by CONTROL-M / Agent for Microsoft Windows.Valid values are: 1 – If a parameter value contains a blank, it is passed to the operating system enclosed in double quotes. If no blank is in the parameter value, no quotes are included. 2 – Parameter values are always passed to the operating system without quotes. If quotes were specified in the job definition, they are removed before the parameter is passed onward by the agent. This option is compatible with the way that these parameters were managed in version 6.0.0x, or 6.1.01 with Fix Pack 1, 2, 3, or 4 installed. In this case, if a parameter value contains a blank, the operating system may consider each string as a separate parameter. 3 – All parameters are passed to the operating system enclosed in double-quotes. This causes the operating system to treat all parameter values as strings (not numbers). This option is compatible with the way that parameters were managed by version 6.1.01 with no Fix Pack installed. 4 – Parameters are passed to the operating system in exactly the same way that they were specified in the job definition. No quotes are added or removed in this case. This option is compatible with the way that parameters were managed by version 2.24.0*x*.
WMI_SETUP_SHARE	Indicates whether the agent will create a shared directory on a remote host. Valid values Y, N. Default: **Y** **Note:** After modifying this parameter, restart CONTROL-M / Agent for the change to take effect.

CONTROL-M Agent services configuration

These parameters affect the operation of the following CONTROL-M / Agent services:

- Agent service
- Tracker service
- FileWatcher service
- Agent Router service

Table 26 System configuration for CONTROL-M/Agent services

Parameter	Description
Log on as	User account under which CONTROL-M Agent and CONTROL-M Tracker services will run. Valid values: **Local System Account**, **This Account**. ■ **Local System Account** – Service will log on as the system account. Default. Subparameter: **Allow Service to Interact with Desktop** – This option can be selected only if the service is running as a local system account. See "Maintaining CONTROL-M / Agent Services" on page 24. ■ **Selected** – the service provides a user interface on a desktop that can be used by whoever is logged in when the service is started. Default. ■ **Unselected** – The service does not provide a user interface. ■ **This Account** – User account under which CONTROL-M Agent and CONTROL-M Tracker services will run. See Logon As User on page 27.
	Note: If the owner of any CONTROL-M / Server jobs has a "roaming profile" or if job output (SYSOUT) will be copied to or from other computers, the Logon mode must be set to **This Account**.
Startup Type	How to install CONTROL-M / Agent and CONTROL-M Tracker services. Valid values: **Automatic**, **Manual**, **Disabled**. Recommended value: **Automatic**. ■ **Automatic** – Services should start when the system starts. Default. ■ **Manual** – User or a dependent service can start services. ■ **Disabled** – User or a dependent service cannot start services.

Defining Windows jobs with advanced parameters

When defining a job for Windows you can define additional job object parameters using the WIN panel of the Job Editing form. Follow the procedures in this chapter to define the additional parameters.

NOTE

Windows jobs using the WIN panel can not be used to run jobs of task type command.

The following tasks are described in this chapter:

Windows job object parameters are described on page 99.

NOTE

For more information about the Job Editing form and Job parameters, see the *CONTROL-M/EM User Guide* and the *CONTROL-M Job Parameter and Variable Reference Guide*.

Creating a job template for Windows jobs

Use this procedure to prepare a template for Windows jobs in CONTROL-M / Desktop.

Before you begin

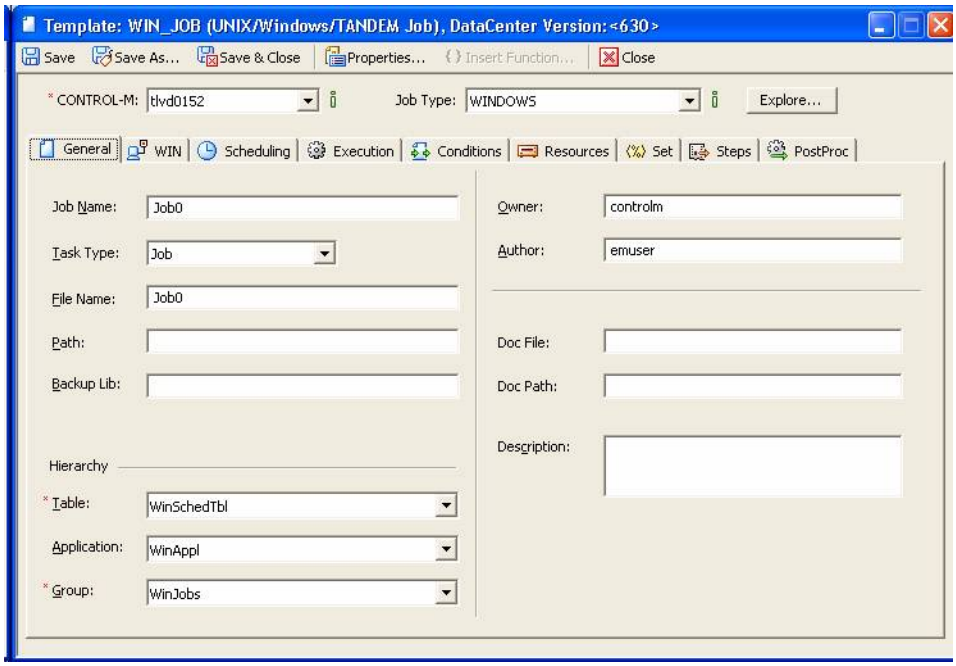
- Ensure that the **windows.xml** file was imported on computers where CONTROL-M/EM client components are installed. For instructions, see the section that deals with Importing WIN Panel Functionality in the *CONTROL-M/Server and CONTROL-M/Agent Installation Guide*.
- To save time, look for a template that resembles one that you want to define, so that the new template can be based on it.

Creating or modifying a template

1 In CONTROL-M/Desktop, open the Template Manager using one of the following methods:

- Choose **Tools => Template Manager**.
- Click **New Job Template** on the toolbar for creating a new template or click **Update** for updating an existing template.

A window similar to the following is displayed:



2 Choose the required **CONTROL-M** from the drop-down list.

3 Choose the required **Job Type** from the drop-down list.

4 Use the fields in this dialog box to create or modify a template.

Where to go from here

Subject	Location
Defining a template	For more information about defining a template, see the *CONTROL-M/EM User Guide*.
Defining a job	For information about defining a job, see "Creating a Windows Job" on page 96 and the *CONTROL-M/EM User Guide*.

Defining the CM in CONTROL-M

Use this procedure to define the CM in CONTROL-M and install the Job Editing form using ctmgetcm.

Before you Begin

- Ensure that the **windows.xml** file was imported on computers where CONTROL-M / EM client components are installed. For instructions, see *Importing WIN Panel Functionality* in the *CONTROL-M/Server and CONTROL-M/Agent Installation Guide*.
- Create a job template. For more information, see "Creating a job template for Windows jobs" on page 93.

To create a CM definition in CONTROL-M

In CONTROL-M/Server

1. Define a group name with application type **WIN**.
2. Specify the CONTROL-M / Agent as a node ID in the node group.
3. Define the <agent_name> in a node group of type WIN (WIN must be in capital letters).
4. Specify ctmgetcm to collect application server information from CONTROL-M / Agent using the following command

```
ctmgetcm -nodeid <agent_name> -appltype WIN -action get
```

In CONTROL-M/EM

1 Install the CONTROL-M/Agent for Microsoft Windows form. For more information, see "Creating a job template for Windows jobs" on page 93.

2 Modify the template according to your requirements. Click **Application** in the Template Manager. See page 93.

3 Click **Load**. The node groups available from CONTROL-M/Server are displayed. Select the required node group.

Where to go from here

Subject	Location
Defining a template	For more information about defining a template, see the *CONTROL-M/EM User Guide*.
Defining a job	For information about defining a job, see "Creating a Windows Job" on page 96. Also see the *CONTROL-M/EM User Guide*.

Creating a Windows Job

Use this procedure to create a new CONTROL-M/Server job processing definition for a Windows job.

Before you begin

- Ensure that the **windows.xml** file was imported on computers where CONTROL-M/EM client components are installed. For instructions, see *Importing WIN Panel Functionality* in the *CONTROL-M/Server and CONTROL-M/Agent Installation Guide*.
- Ensure that a job template was defined for Windows jobs. For more information, see "Creating a job template for Windows jobs" on page 93.
- The job name of the Windows job you want to create must contain only characters (letters, digits, underscore) that can be used in a Microsoft Windows file name.

NOTE

When CONTROL-M/Agent manages a large number of jobs on a computer with Microsoft Windows 2000 Professional, the following message may be displayed "**Application error 142**".

To create a Windows job

1 Choose one of the following:

- **Edit => New Job** from the menu bar
- Press **Ctrl + J**
- Click  .

2 Select a template for Windows jobs from the **Apply Template** drop-down list in CONTROL-M / Desktop.

NOTE

For descriptions of all panels in the Job Editing form, see the *CONTROL-M/EM User Guide*.

3 Click the **WIN** tab of the Job Editing form.

The WIN panel is used to set parameter values for some Windows jobs. This process is described in "Setting Job Object limits for a Windows job" on page 97.

NOTE

For more information about the Job Editing form and Job Editing parameters, see the *CONTROL-M/Enterprise Manager User Guide*.

Setting Job Object limits for a Windows job

Use this procedure to specify values for job object limitation parameters in the WIN panel of the Job Editing Form. If a value is not specified for a parameter, that parameter does not limit the job in any way.

NOTE

Job Object features are not supported in jobs processed on a remote host.

Before you begin

The **WIN2K** template is used for defining job object limitations. If this template does not exist, see "Creating a job template for Windows jobs" on page 93.

To view the characteristics of the job object whose attributes are to be specified, see "Viewing a Job Object" on page 100.

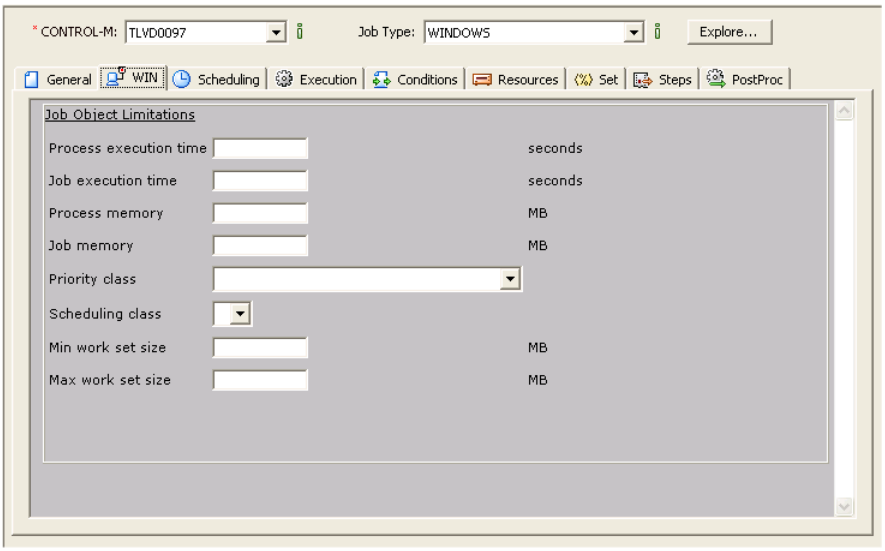
NOTE

In messages, "job object" may be abbreviated as **JO**.

To set Job Object limits for a Windows job

1 Click the **WIN** tab of the Job Editing form. The WIN Panel is displayed.

Figure 9 WIN Panel



WIN panel parameters are described in Table 27 on page 99. All these parameters are optional.

2 Specify new data or modify existing data in the WIN Panel.

3 When you are satisfied with the data in the WIN panel, click **Save to Draft** to save the parameter specifications.

Where to go from here

The following table describes where you might find information about tasks that are related to this one.

Subject	Location
Defining additional Windows jobs	You can now define additional Windows jobs by clicking the WIN tab of the Job Editing form.
Viewing jobs you have created or modified	Click **Cancel** to exit the WIN Panel and go to the Active environment to view the jobs you have created or modified.
Defining jobs	For information about defining jobs, see the *CONTROL-M/EM User Guide*.

Windows job object parameters

WIN panel parameters are described in Table 27. This table also contains the names of the corresponding AutoEdit variables that can be included on the command line or in a parameter input file when invoking the CONTROL-M/Server ctmcreate or ctmdefine utilities. For example,

```
-autoedit %%WIN2K-PRIORITY_CLASS = "ABOVE_NORMAL_PRIORITY_CLASS"
```

NOTE

Values entered in WIN panel fields are validity checked. Values entered on the command line or in a parameter input file are not validity checked.

Except for Priority class and Scheduling class, all of these parameters can have decimal values (real numbers). Unless stated otherwise, maximum and minimum values are computer dependent.

Table 27 Windows job object parameters

WIN Panel Parameter	Definition and AutoEdit Variable for CONTROL-M/Server
Process execution time	Maximum CPU time, in seconds, for each process in a job. Valid values: Minimum: **0.1**. Maximum: **1.8 x 1012** **%%WIN2K-PER_PROCESS_USER_TIME_LIMIT**
Job execution time	Maximum CPU time, in seconds, for entire job. Valid values: Min: **0.1**. Max: **1.8 x 1012** **%%WIN2K-PER_JOB_USER_TIME_LIMIT**
Process memory	Maximum memory, in megabytes, allowed for each process in a job. Valid values: Minimum: **0.1**. Maximum: **4200.0**. **%%WIN2K-PROCESS_MEMORY_LIMIT**
Job memory	Maximum memory, in megabytes, allowed for job. Valid values: Min: **0.1**. Max: **4200.0** **%%WIN2K-JOB_MEMORY_LIMIT**

Table 27 Windows job object parameters

WIN Panel Parameter	Definition and AutoEdit Variable for CONTROL-M/Server
Priority class	Highest priority class the job and its "children" can receive. Valid values: ■ IDLE_PRIORITY_CLASS ■ BELOW_NORMAL_PRIORITY_CLASS ■ NORMAL_PRIORITY_CLASS ■ ABOVE_NORMAL_PRIORITY_CLASS ■ HIGH_PRIORITY_CLASS ■ REALTIME_PRIORITY_CLASS **%%WIN2K-PRIORITY_CLASS**
Scheduling class	Scheduling class for all processes of a job. Valid values: **0** – **9**. ■ **0** - provides the minimum resources ■ **9 -** provides the maximum resources. **%%WIN2K-SCHEDULING_CLASS**
Minimum Working set size	Minimum RAM, in megabytes, for all processes of the job. Increasing the value of this parameter reduces page swapping for this process but reduces the RAM available for other processes. Valid values: Min.: **0.1**. Max.: **4200.0** **%%WIN2K-MINIMUM_WORKING_SET_SIZE**
Maximum Working set size	Maximum RAM, in megabytes, for all job processes. Decreasing this value reduces the likelihood that this job will interfere with other jobs but may increase the execution time for this job. Valid values: Min.: **0.1**. Max.: **4200.0** **%%WIN2K-MAXIMUM_WORKING_SET_SIZE**

NOTE

You can determine the amount of memory and CPU time required for a job by viewing job object statistics in the job SYSOUT. If a job exceeds job object limitations, the job will end NOTOK and the same error message will appear in the SYSOUT and in the CONTROL-M/Server message log. WIN panel parameters are used in CONTROL-M/Agent by the SetInformationJobObject API. For more information, see Microsoft documentation for the SetInformationJobObject API.

Viewing a Job Object

This Microsoft Windows procedure enables you to determine job object characteristics of any CONTROL-M/Server job. You can use these characteristics to decide what job object limits should be specified.

Before you Begin

You can only view Job Object details for a job that is currently running. Use the Order or Force option to run the job you want to evaluate.

To view a Job Object

1 Choose **... => Administrative Tools => Performance**.

2 The **Performance** window (including its graph panel) is displayed. Click the large + button.

3 The **Add counters** dialog box is displayed.

A In the **Performance object** drop-down list, select **JobObject**.

B Choose **All instances** to display all existing job object instances.
or
Choose **Select instances from list** and select one or more of the displayed instances.

The job instance format is <job_name> <order_number> <run_number>. These elements are separated by a blank.

C Choose **Select counters from list**. Then, select the characteristic in the displayed list that you want to analyze.

To view objects on other Microsoft Windows computers, select counters from <name_of_computer> in the Add counters dialog box.

D Click **Add**.

4 Data for the selected "counter" is displayed in the **Performance** graph. Analyze this data to determine whether to specify a related job object limit and, if so, the limiting value to specify.

Where to go from here

The following table describes where to find information about tasks that are related to this one.

Subject	Location
Viewing a Job Object	If you need to view other characteristics for the same job object or view the characteristics of other job objects, repeat Steps 3 and 4 of this procedure.
Setting Job Object limits	Use the procedure on page 97 to specify values for the parameters in the **WIN** tab of the Job Editing form.
Defining additional Windows jobs	You can create additional Windows jobs by clicking the **WIN** tab of the Job Editing form. For information about defining jobs, see the *CONTROL-M/EM User Guide*.
Viewing jobs you have created or modified	Click **Cancel** in the **WIN** tab the Job Editing form to go to the Active environment to view the jobs you have created or modified.

Index

Symbols

A

B

C

D

E

F

H

I

J

L

M

O

P

R

S

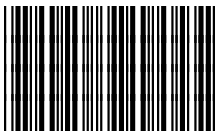
T

U

V

W

Notes



63762