



Maintaining the Central Management System Database

This chapter describes how to maintain the Central Management System (CMS) database using CLI commands as well as using the Content Distribution Manager GUI.

This chapter contains the following sections:

- [Scheduling a CMS Maintenance Routine Using the CLI, page 12-1](#)
- [Scheduling a CMS Maintenance Routine Using the Content Distribution Manager GUI, page 12-3](#)



Note

The CMS database maintenance runs at the scheduled time only when the following three conditions are satisfied:

- The last vacuum process happened more than 30 minutes in the past.
- The percent increase in disk space usage is greater than 10 percent.
- The available free disk space is greater than 10 percent of the total disk space.

If any of these conditions are not satisfied, the CMS database maintenance does not run at the scheduled time, and the **show cms database maintenance** EXEC command output states, “Database maintenance is not running.”

Scheduling a CMS Maintenance Routine Using the CLI

To schedule maintenance of the CMS database, use the **cms database maintenance** global configuration command. You can do a routine full maintenance cleaning (“vacuuming”) or a regular maintenance reindexing of the embedded database. The full maintenance routine runs only when the disk is more than 90 percent full and runs only once a week. When you clean the tables, the reusable space is returned to the database system.

The **cms enable** command automatically registers the node in the database management tables and enables the CMS. The **no cms enable** command only stops the management services on the device and does not disable a primary sender. You can use the **cms deregister** command to remove a primary or backup sender Content Engine from the ACNS network and to disable communication between the two multicast senders.

The syntax of the **cms database maintenance** command is as follows:

```
cms {database maintenance {full {enable | schedule weekday at time} | regular {enable | schedule
weekday at time}} | enable | rpc timeout {connection 5-1800 | incoming-wait 10-600 | transfer
10-7200}
```

Table 12-1 describes the syntax elements.

Table 12-1 *cms database maintenance Command Syntax Description*

Element	Description
database maintenance	Configures the embedded database clean or reindex maintenance routine.
full	Configures the full maintenance routine and cleans the embedded database tables.
enable	Enables the full maintenance routine to be performed on the embedded database tables.
schedule	Sets the schedule for performing the maintenance routine.
<i>weekday</i>	Day of the week to start maintenance routine. every-day: Every day Mon: every Monday Tue: every Tuesday Wed: every Wednesday Thu: every Thursday Fri: every Friday Sat: every Saturday Sun: every Sunday
at	Sets the maintenance schedule time of day to start maintenance routine.
<i>time</i>	Time of day to start maintenance routine (0–23:0–59)(hh:mm).
regular	Configures the regular maintenance routine and reindexes the embedded database tables.
enable	Enables the node CMS process.
rpc timeout	Configures the timeout values for remote procedure call connections.
connection	Specifies the maximum time to wait when making a connection.
<i>5-1800</i>	Timeout period in seconds. The default for the Content Distribution Manager is 30 seconds; for the Content Engine and the Content Router, it is 180 seconds.
incoming-wait	Specifies the maximum time to wait for a client response.
<i>10-600</i>	Timeout period in seconds. The default is 30 seconds.
transfer	Specifies the maximum time to allow a connection to remain open.
<i>10-7200</i>	Timeout period in seconds. The default is 300 seconds.

The following example schedules a regular (reindexing) maintenance routine to start every Friday at 11:00 at night:

```
ContentEngine(config)# cms database maintenance regular schedule Fri at 23:00
```

The following example shows how to enable the CMS process on a Content Engine:

```

ContentEngine(config)# cms enable
This operation needs to restart http proxy and streaming proxies/servers (if running) for
memory reconfiguration. Proceed? [no]yes
Registering this node as Content Engine...
Thu Jun 26 13:18:24 UTC 2003 [I] main: creating 24 messages
Thu Jun 26 13:18:25 UTC 2003 [I] main: creating 12 dispatchers
Thu Jun 26 13:18:25 UTC 2003 [I] main: Sending registration message to CDM 10.107.192.168
Thu Jun 26 13:18:27 UTC 2003 [I] main: Connecting storeSetup for CE.
Thu Jun 26 13:18:27 UTC 2003 [I] main: Instantiating AStore
'com.cisco.unicorn.schema.PSqlStore'...
Thu Jun 26 13:18:28 UTC 2003 [I] main: Successfully connected to database
Thu Jun 26 13:18:28 UTC 2003 [I] main: Registering object factories for persistent
store...
Thu Jun 26 13:18:35 UTC 2003 [I] main: Dropped Sequence IDSET.
Thu Jun 26 13:18:35 UTC 2003 [I] main: Dropped Sequence GENSET.
Thu Jun 26 13:18:35 UTC 2003 [I] main: Dropped Table USER_TO_DOMAIN.
.
.
.
Thu Jun 26 13:18:39 UTC 2003 [I] main: Created Table FILE_CDM.
Thu Jun 26 13:18:40 UTC 2003 [I] main: Created SYS_MESS_TIME_IDX index.
Thu Jun 26 13:18:40 UTC 2003 [I] main: Created SYS_MESS_NODE_IDX index.
Thu Jun 26 13:18:40 UTC 2003 [I] main: No Consistency check for store.
Thu Jun 26 13:18:40 UTC 2003 [I] main: Successfully created management tables
Thu Jun 26 13:18:40 UTC 2003 [I] main: Registering object factories for persistent
store...
Thu Jun 26 13:18:40 UTC 2003 [I] main: AStore Loading store data...
Thu Jun 26 13:18:41 UTC 2003 [I] main: ExtExpiresRecord Loaded 0 Expires records.
Thu Jun 26 13:18:41 UTC 2003 [I] main: Skipping Construction RdToClusterMappings on
non-CDM node.
Thu Jun 26 13:18:41 UTC 2003 [I] main: AStore Done Loading. 336
Thu Jun 26 13:18:41 UTC 2003 [I] main: Created SYS_MESS_TIME_IDX index.
Thu Jun 26 13:18:41 UTC 2003 [I] main: Created SYS_MESS_NODE_IDX index.
Thu Jun 26 13:18:41 UTC 2003 [I] main: No Consistency check for store.
Thu Jun 26 13:18:41 UTC 2003 [I] main: Successfully initialized management tables
Node successfully registered with id 28940
Registration complete.
Warning: The device will now be managed by the CDM. Any configuration changes made via CLI
on this device will be overwritten if they conflict with settings on the CDM.
Please preserve running configuration using 'copy running-config startup-config'.
Otherwise management service will not be started on reload and node will be shown
'offline' in CDM UI.
management services enabled
ContentEngine(config)#

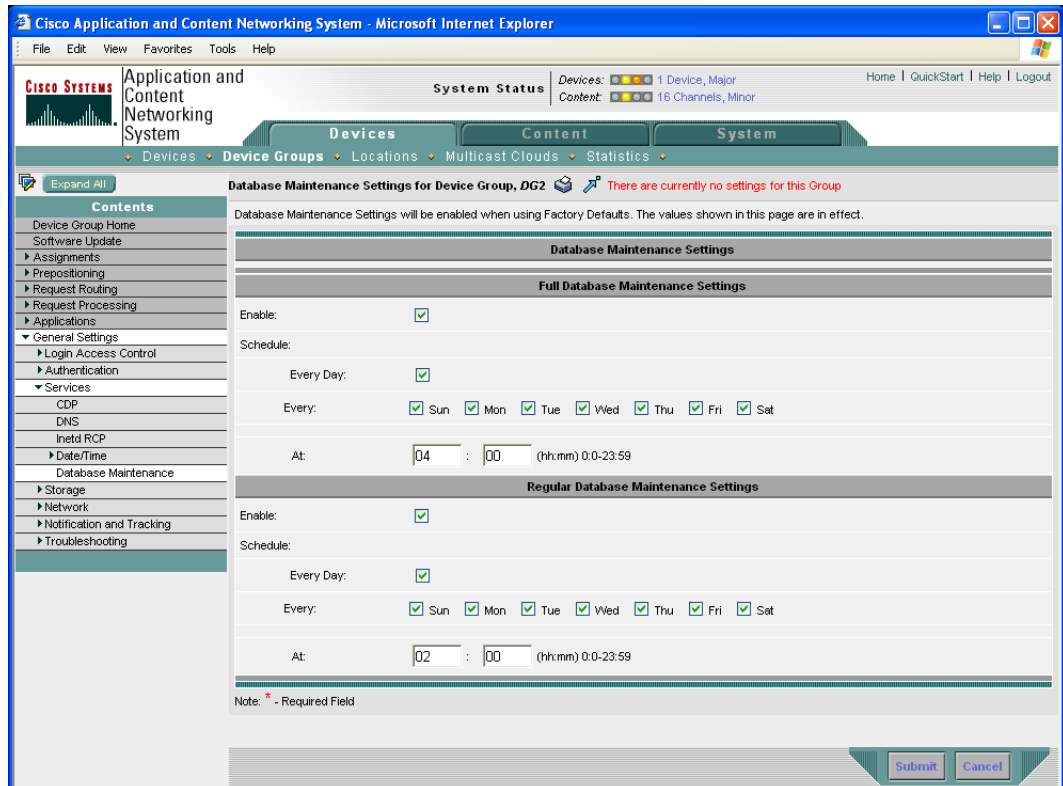
```

Scheduling a CMS Maintenance Routine Using the Content Distribution Manager GUI

To schedule a CMS database cleaning or reindexing using the Content Distribution Manager GUI (available as of the ACNS 5.3 Release), follow these steps:

- Step 1** From the Content Distribution Manager GUI, choose **Devices > Devices** (or **Device Groups**).
- Step 2** Click the **Edit** icon next to the Content Engine or device group for which you want to schedule a CMS maintenance routine.
- Step 3** In the Contents pane, choose **General Settings > Services > Database Maintenance**. The Database Maintenance Settings window appears. (See [Figure 12-1](#).)

Figure 12-1 Database Maintenance Settings for Device Group Window



- Step 4** In the Full Database Maintenance Settings area, click the **Enable** check box to carry out a full database maintenance routine.
- Step 5** Choose **Every Day** to carry out the full database maintenance schedule every day.
If you choose **Every Day**, then the field Every and the fields pertaining to the days of the week are all chosen.
- Step 6** If you have not chosen **Every Day**, then choose the days of the week on which you want the full database maintenance.
If you choose a particular day of the week, the full database maintenance routine is carried out on that particular day every week.
You can choose multiple days of the week to carry out the full database maintenance routine.
- Step 7** Enter the time at which the full database maintenance routine is to be carried out.
Enter the time as hh:mm in 24-hour format.
- Step 8** In the Regular Database Maintenance Settings area, choose **Enable** to carry out database reindexing.
- Step 9** Choose **Every Day** to reindex the database every day.
If you choose **Every Day**, then the field Every and the fields pertaining to the days of the week are all chosen.
- Step 10** If you have not chosen **Every Day**, then choose the days of the week on which you want the database reindexed.
If you choose a particular day of the week, the database is reindexed on that particular day every week.
You can choose multiple days of the week on which the database is to be reindexed.

- Step 11** Enter the time at which the database reindexing is to be carried out.
Enter time as hh:mm in 24-hour format.
- Step 12** Click **Submit** to save the settings.

Table 12-2 describes the fields in the Database Maintenance Settings window.

Table 12-2 Database Maintenance Settings

Field	Description
Full Database Maintenance Settings	
Enable	Full database maintenance routine is enabled.
Every Day	Full database maintenance is to be carried out every day. When Every Day is chosen, the field Every and the fields pertaining to the days of the week are all chosen.
Every	The full database maintenance routine is carried out every week on the specified days (along with the relevant days of the week fields).
At	Time at which the full database maintenance routine is to be carried out. Enter time as hh:mm in 24-hour format.
Regular Database Maintenance Settings	
Enable	Database reindexing is enabled.
Every Day	Database is reindexed every day. When Every Day is chosen, the field Every and the fields pertaining to the days of the week are all chosen.
Every	The database is reindexed every week on the specified days (along with the relevant days of the week fields).
At	Time at which the database is to be reindexed. Enter time as hh:mm in 24-hour format.

