



Services and Processes

- [Services, on page 1](#)
- [Using the Local Desktop, on page 9](#)
- [ICM Service Control and Windows Task Manager, on page 9](#)
- [Using the Local Registry, on page 9](#)
- [Using the Remote SNMP Management Station, on page 10](#)

Services

The following table lists the processes running on a particular server. In the Description column, the criticality of a process is shown within brackets []. The key definitions are as follows:

Critical

This process is critical to the operation of the component. Failure of the process renders the application either dysfunctional or impaired.

Critical/Optional

This process is optional (needed for a feature often enabled via configuration or during product installation). However, if the feature is enabled, the process is critical and failure of the process is likely to render the application either dysfunctional or impaired.

Optional

This process is optional (needed for a feature often enabled via configuration or during product installation). Failure of the process is unfortunate but does not impair the Contact Center application.

Important

While failure of this process does not impair the Contact Center application, it disables an important capability.

Non-Critical

This process runs on the server under normal operating conditions, but its failure has little or no impact on the Contact Center application.

An asterisk preceding the process name denotes that this process appears in the SNMP CISCO-CONTACT-CENTER-APPS-MIB cccaComponentElmtTable.

Table 1: Unified ICM/Unified CCE Processes

| Component | Process | Description |
|-----------|----------------|--|
| Router | * router.exe | [Critical] This is the primary Router process. |
| | * rtsvr.exe | [Critical] Provides real-time data feed from the Router to the Administration & Data Server. |
| | * mdsproc.exe | [Critical] Message Delivery Service |
| | * ccagent.exe | [Critical] Router component that manages communication links between the Router and peripheral gateways. |
| | * dbagent.exe | [Critical] Manages connections and transactions (configuration updates) from configuration tools. |
| | * testsync.exe | [Non critical] Provides interface for component test tools. |
| | * appgw.exe | [Optional/Critical] The process that provides an interface for the Router to communicate with external applications. |
| | * dbworker.exe | [Optional/Critical] The process that provides the interface for the Router to query external databases. |
| | * [NIC].exe | [Optional/Critical] A separate process is active for each Network Interface Controller (NIC) enabled during SETUP. The NIC process manages the interface to a telephony network. The presence of a NIC process in a Unified CCE deployment is very rare. NIC process names: atnnc.exe, cainnc.exe, netwrkcic.exe, crspnc.exe, gktmpnc.exe, incrpnc.exe, mcinnc.exe, gennnc.exe, ntnnc.exe, ntlnc.exe, sprnc.exe, ss7innnc.exe, stentornnc.exe, timnnc.exe, unisourcenc.exe |

| Component | Process | Description |
|-----------|-----------------------|--|
| Logger | * configlogger.exe | [Critical] The process that manipulates configuration data. |
| | * histlogger.exe | [Critical] The process that inserts historical data into TMP historical tables in the Logger database. |
| | * recovery.exe | [Critical] This process bulk copies historical data from the TMP historical tables to the actual historical tables. Recovers and synchronizes historical data with its partner logger during failover if loggers are running duplex. It is also responsible for historical data purges in the Logger database based on configured retention parameters. |
| | * replication.exe | [Critical] The process that replicates data from the Logger to the Historical Data Server on an Administration & Data Server. |
| | * csfs.exe | [Critical] The alarm/event processor. CSFS distributes alarms/events send via EMS to supported alarm/event feeds, for example, SNMP, syslog. CSFS stands for Customer Support Forwarding Service, which in Unified ICM's infancy, forwarded events to a central monitoring location. |
| | * cw2kfeed.exe | [Optional] The syslog event feed. This process acquires events from the CSFS process, formats them appropriately in accordance with the syslog protocol and sends the events to the configured collector. If a syslog collector is not configured, the event feed from CSFS will not be processed. |
| | * campaignmanager.exe | [Optional/Critical] Outbound Option Campaign Manager. This process manages customer lists: provides customer records for every Dialer in the enterprise; determines when customers should be called again; maintains the "Do Not Call" list in memory. The Campaign Manager also sends real time and historical data to the Router and distributes configuration information to Dialer and Import processes. |
| | * baimport.exe | |

| Component | Process | Description |
|-----------|--------------|--|
| | | [Optional/Critical] Outbound Option Import process. This process imports contact lists into the Outbound Option database; applies query rules to the contact table to build dialing lists; determines the GMT value for each phone based on the region prefix configuration. |
| | sqlservr.exe | [Critical] Microsoft SQL server process |
| | sqlagent.exe | [Critical] Microsoft SQL server process |

| Component | Process | Description |
|-----------|----------------|---|
| PG | * opc.exe | [Critical] Open Peripheral Controller (OPC). This process acts as the brain for the peripheral gateway, including acting as a central collection and distribution point for all interaction with peripherals. OPC also ensures that all synchronization is accomplished with the other side. It also prepares and sends termination call detail (TCD) records as well as 5 minute and 30 minute reports. |
| | * mdsproc.exe | [Critical] Message Delivery Service |
| | * pgagent.exe | [Critical] MDS Peripheral Gateway component that manages the interface between the peripheral gateway and the central controller. |
| | * testsync.exe | [Non critical] Provides interface for component test tools. |
| | * eagtpim.exe | [Optional/Critical] The Cisco Unified CM peripheral interface manager process. This process manages the interface between OPC and the JTAPI Gateway. Multiple PIMs of the same type can be enabled for a PG. VRU PIMs and Unified CM PIMs may be coresident on a PG as well. This is very common in Unified CCE deployments but may not be present on all PGs. There may be multiple instances of this process running. |
| | * acmipim.exe | [Optional/Critical] The process is expected on the Unified SCCE Gateway PG – this Peripheral Interface Manager is responsible for the communication interface between the parent instance and the child instance. |
| | * vrupim.exe | [Optional/Critical] Peripheral Interface Manager process between OPC and a Voice Response Unit (VRU) or Interactive Voice Response (IVR). There may be multiple instances of this process running. |
| | * mrpim.exe | |

| Component | Process | Description |
|-----------|-----------------------|--|
| | | <p>[Optional/Critical] The Media Routing Peripheral Interface Manager is the integration point for the Outbound Option Dialer, Cisco Email Manager (CEM), Cisco Collaboration Server (CCS) and the Enterprise Chat and Email.</p> <p>There may be multiple instances of this process running.</p> |
| | * msgis.exe | <p>[Optional/Critical] Message Integration Service (MIS), which provides a mechanism to share call context data with a VRU. This process is only present on a PG with a VRU PIM.</p> |
| | * ctiosservernode.exe | <p>[Critical] The CTI OS Server process that manages connections from CTI clients (agent desktops), retains (real-time) data about agents and acts as the conduit for events and control messaging between CTI Server and CTI clients.</p> |
| | * jtapigw.exe | <p>[Critical] JTAPI Gateway that manages the interface to the Unified Communications Manager IP PBX via the JTAPI client to the CTI Manager on the Unified CM. On the other side, the JTAPI Gateway connects to the Unified CM PIM and translates JTAPI messages and events into a format expected by the PIM.</p> |
| | * ctisvr.exe | <p>[Critical] CTI Gateway (CTI Server) process that processes (GED-188) messages between CTI OS and OPC.</p> <p>Note: In legacy implementations, CTI Server manages connections to CTI desktops.</p> |

| Component | Process | Description |
|---------------------------------------|---|---|
| Administration & Data Server (AW/HDS) | * configlogger.exe | [Critical] Processes inbound configuration data. |
| | * updateaw.exe | [Critical] Updates the local configuration database with configuration data from the central controller. |
| | * rtclient.exe | [Critical] Receives a real-time data feed (from a real-time distributor) and updates the local database. |
| | * rtdist.exe | [Critical] Manages inbound real-time data from the real time server on the Router and distributes it to real-time clients. |
| | * replication.exe | [Critical] Manages replicated historical data received from the Logger (HDS only) and inserts historical data in the HDS database. In addition, it is responsible for historical data purges in the HDS database based upon configured retention parameters. |
| | * cmsnode.exe | <p>[Optional] Configuration Management System (CMS). Manages configuration data for the ConAPI interface. This is a necessary interface (process) for the System CCE web configuration. Thus, for System Unified CCE, this is an important process. Also, if the customer has purchased the Cisco Unified Contact Center Management Portal (Unified CCMP), CONAPI is also used. However, for a Unified CCE deployment without Unified CCMP, this process is not critical.</p> <p>In a recent version of Unified CCE, cmsnode.exe runs by default but it is difficult for a management station to know whether it is necessary. Therefore, this is listed as Optional.</p> |
| * cms_jserver.exe | <p>[Optional] Configuration Management System (CMS) Jaguar Server. This process works with cmsnode.exe for CMS to provide Java interfaces for ConAPI.</p> <p>In a recent version of Unified CCE, cms_jserver.exe runs by default but it is difficult for a management station to know whether it is necessary. Therefore, this is listed as Optional.</p> | |

| Component | Process | Description |
|-----------|-----------------------|--|
| | tomcat<version>.exe | [Optional/Critical] Apache Tomcat servlet engine for SCCE web config. |
| | * iseman.exe | [Optional] Internet Script Editor |
| | sqlservr.exe | [Critical] Microsoft SQL server process |
| | sqlagent.exe | [Optional] Microsoft SQL server process |
| All Nodes | nodeman.exe | [Critical] Node Manager. This process monitors the status of all Unified ICM/Unified CCE processes on the server; should a process terminate unexpectedly, the Node Manager automatically restarts that process. |
| | nmm.exe | [Critical] Node Manager Manager. This process monitors the primary Node Manager (nodeman.exe) process; should the primary Node Manager (nodeman.exe) process terminate unexpectedly, the Node Manager Manager restarts it. |
| | snmpdm.exe | [Important] SNMP master agent. |
| | cccsnmpgmt.exe | [Important] SNMP agent management service – this service manages the SNMP agent infrastructure and restarts any agents that may terminate unexpectedly. It also ensures that the agent processes run at a reduced priority so as to not adversely impact server performance. |
| | msnsagt.exe | [Important] Microsoft native subagent adapter. |
| | UcceSnmpHelperX86.exe | [Important] Used by hostagt.exe and cccaagent.exe to get information about 32-bit processes that are currently running on the machine. |
| | hostagt.exe | [Important] HOST-RESOURCES-MIB subagent. |
| | sappagt.exe | [Important] SYSAPPL-MIB subagent. |
| | cccaagent.exe | [Important] CISCO-CONTACT-CENTER-APPS-MIB subagent. |

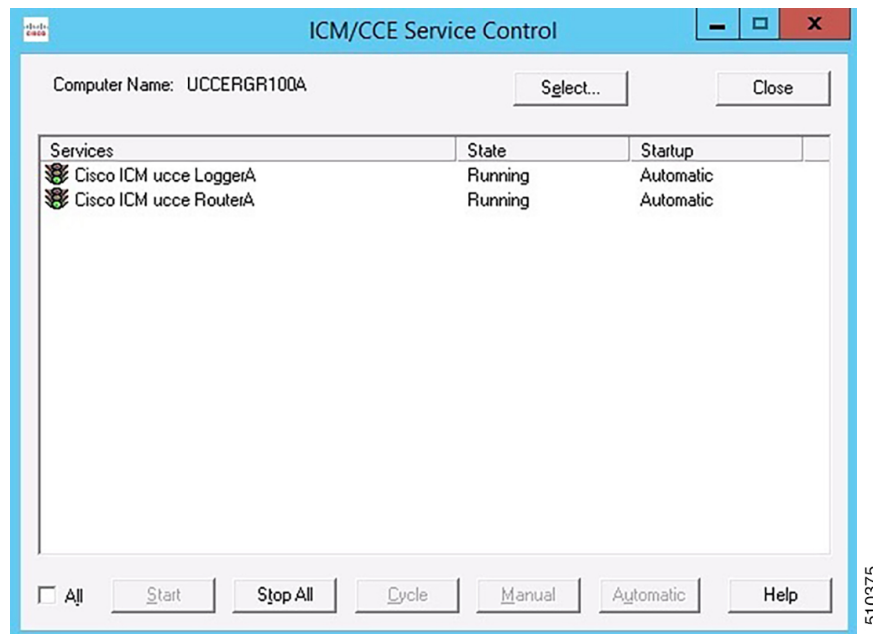
Using the Local Desktop

Use the Unified ICM Service Control and the local registry to monitor Unified ICM/Unified CCE components and their processes.

ICM Service Control and Windows Task Manager

The Unified ICM Service Control displays the Node Manager service for each Unified ICM/Unified CCE component and its state and startup settings. Each Node Manager service appears in the following format: Cisco ICM <instance> <component>. The following Unified ICM Service Control window example lists information about the Node Manager services running on the local machine. The Router component Node Manager service is identified as “Cisco ICM acme RouterA”.

Figure 1: ICM Service Control



On Windows 2008, you can no longer view Unified ICM/Unified CCE processes on the Application tab of Windows Task Manager. To view the status of each process, use the Diagnostic Framework Portico. For more information, see [Accessing the Diagnostic Framework Through the Built-In User Interface \(Portico\)](#).

Using the Local Registry

The Unified ICM/CCE Windows registry hive contains the set of all installed components and their processes. However, to determine which processes are being managed, traverse the Node Manager registry key for each component.

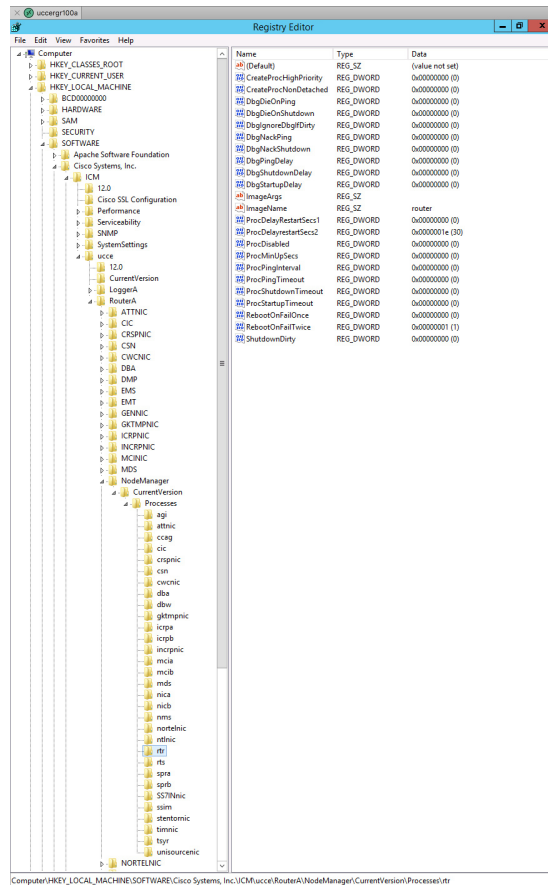
The following illustration shows the set of processes associated with the Cisco ICM acme RouterA component. The key name for the Router process is rtr; it appears highlighted in the navigation pane of the Registry Editor

window. The process name, Router, is contained in the ImageName value; it appears without the .exe file extension. If the ProcDisabled value is set to 0—as is the case for the Router process—the RouterA Node Manager process starts and manages the process.



Note The key name is typically not the same as the process name.

Figure 2: Registry Editor



510374

Using the Remote SNMP Management Station

In addition to the information available using the local desktop tools and registry, the Contact Center SNMP agent returns information about all Unified ICM/Unified CCE-enabled processes regardless of whether they are running. This information is available from the `cccaInstanceTable`, `cccaComponentTable`, and `cccaComponentElmtTable`. The instance number and component index correlate a process to a specific instance and component.

The first example shows the entries for acme-RouterA Router process. The `cccaComponentElmtRunID` value, which is the process ID, is valid if the `cccaComponentElmtStatus` is active, started, or standby.

```
cccaInstanceName.0 = acme
cccaComponentType.0.1 = router(1)
```

```
cccaComponentName.0.1 = RouterA
cccaComponentStatus.0.1 = started(4)
cccaComponentElmtName.0.1.5 = router
cccaComponentElmtRunID.0.1.5 = 4040
cccaComponentElmtStatus.0.1.5 = active(5)
```

The next example shows the entries for acme-LoggerA, the configlogger process. The `cccaComponentElmtRunID` value, which is the process ID, is valid if the `cccaComponentElmtStatus` is not stopped (3).

```
cccaInstanceName.0 = acme
cccaComponentType.0.2 = logger(2)
cccaComponentName.0.2 = LoggerA
cccaComponentStatus.0.2 = stopped(3)
cccaComponentElmtName.0.2.8 = configlogger
cccaComponentElmtRunID.0.2.8 = 0
cccaComponentElmtStatus.0.2.5 = stopped(3)
```

