



Cisco JTAPI Installation and Configuration

Cisco Java Telephony API (JTAPI) implementation consists of Java classes that reside on all client machines running Cisco JTAPI applications. Installation of the Cisco JTAPI implementation is necessary before Cisco JTAPI applications will function correctly. The Cisco JTAPI classes are installed wherever JTAPI applications will run, whether on Cisco CallManager, on a separate machine, or on both. Installation requires 5 MB of local disk space.

This section discusses the software installation and configuration required to ready the Cisco JTAPI environment for programming and running applications.

- Installing the Cisco JTAPI Software, page B-2
- Verifying the Installation, page B-4
- Configuring Cisco JTAPI Tracing, page B-4
- Administering User Information for JTAPI and TAPI Applications, page B-14

Installing the Cisco JTAPI Software

Perform the steps below to install the CiscoJTAPI software:

Procedure

- Step 1** Log in to the computer where you want to install the Cisco JTAPI client software.
- Step 2** Close all Windows programs.
- Step 3** Open a web browser.
- Step 4** Go to the Cisco CallManager administration pages:
<http://Name/CCMAdmin/main.asp>
where:
Name is the name or IP address of the Cisco CallManager
- Step 5** Click **Application > Install Plugins**.
- Step 6** Click the **Cisco JTAPI** link.
- Step 7** Save the file on your desktop or run it from the download site.
- Step 8** Follow the instructions in the pop-up windows.
-



Note

Install Cisco JTAPI software on the default drive as directed by the installation software. The default directory, for example, is C:\WINNT\Java\lib, when Windows NT is installed in C:\WINNT.

Installed files

The cisco JTAPI installation utility installs the following items on the local disk drive:

- JTAPI java classes in %SystemRoot%\java\lib
- JTAPI Preferences (jtprefs.exe) in Program Files\JTAPITools
- JTAPI sample applications(makecall, muzak, jtrace) in Program Files\JTAPITools
- JTAPI documentation in Program Files\JTAPITools

**Note**

To run many JTAPI applications, all you need is a Java 1.1-compatible environment such as the Sun JDK 1.1.x or Microsoft Virtual Machine (bundled with Internet Explorer 4.0 and higher). Cisco JTAPI will also run on Sun JDK 1.2.

To use JTPREFS (application to set Cisco JTAPI tracing parameters) and the sample applications bundled with Cisco JTAPI, however, Microsoft Java Virtual Machine 5.00.3190 or later is required. If you are installing Cisco JTAPI on a Windows 2000 Workstation or Server, you already have a compatible version of the Microsoft Virtual Machine. On all other Microsoft platforms such as Windows 95, Windows 98, and Windows NT, verify the current version of the Microsoft Virtual Machine by running the command "jview /?" and noting the version printed at the top of the console. You can download the latest version of the Microsoft Java Virtual Machine from <http://www.microsoft.com/java>.

Verifying the Installation

When you perform the procedure in this section, a call is placed in order to test the Cisco JTAPI installation and configuration. Perform the following steps:

Procedure

Step 1 From the Windows NT command line, navigate to the directory where you installed Cisco JTAPI Tools. By default, this directory is Program Files\JTAPITools.

Step 2 Execute the following command:

Jview makecall <server name> <login> <password> 1000 <phone1> <phone2>
where

server name is the hostname or IP address of the Cisco CallManager (for example, CTISERVER).

phone1 and *phone2* are directory numbers of IP phones or virtual phones that are controlled by the user according to the user configuration. Refer to the “Adding a New User” section for details.

login and *password* are the user ID and password that you configured in the Cisco CallManager User Configuration page.

Configuring Cisco JTAPI Tracing

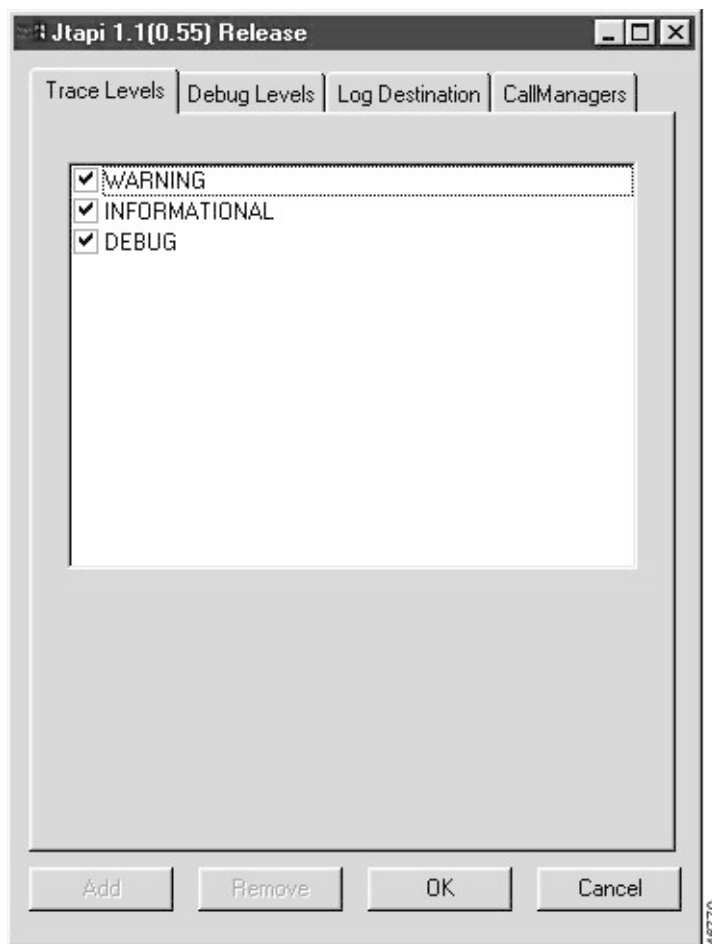
You use the Cisco JTAPI tracing preferences application (JTPREFS.EXE) to configure trace levels, trace destinations, and CallManager names. The Cisco JTAPI Preferences utility is installed by default in the directory Program Files\JTAPITools. To open Cisco JTAPI preferences, select **Start > Programs > Cisco JTAPI**.

Trace Levels

This section describes how to use the Cisco JTAPI preferences application. The following topics are discussed:

- Debug Levels, page B-7
- Log Destination, page B-8
- Cisco CallManager, page B-11
- Other JTAPI Preferences Settings, page B-12
- JTAPI Preferences on non-Microsoft environments, page B-13

Figure 0-1 illustrates the trace level window of the Cisco JTAPI preferences application. The JTAPI version number is shown in the window title.

Figure 0-1 Cisco JTAPl Trace Levels

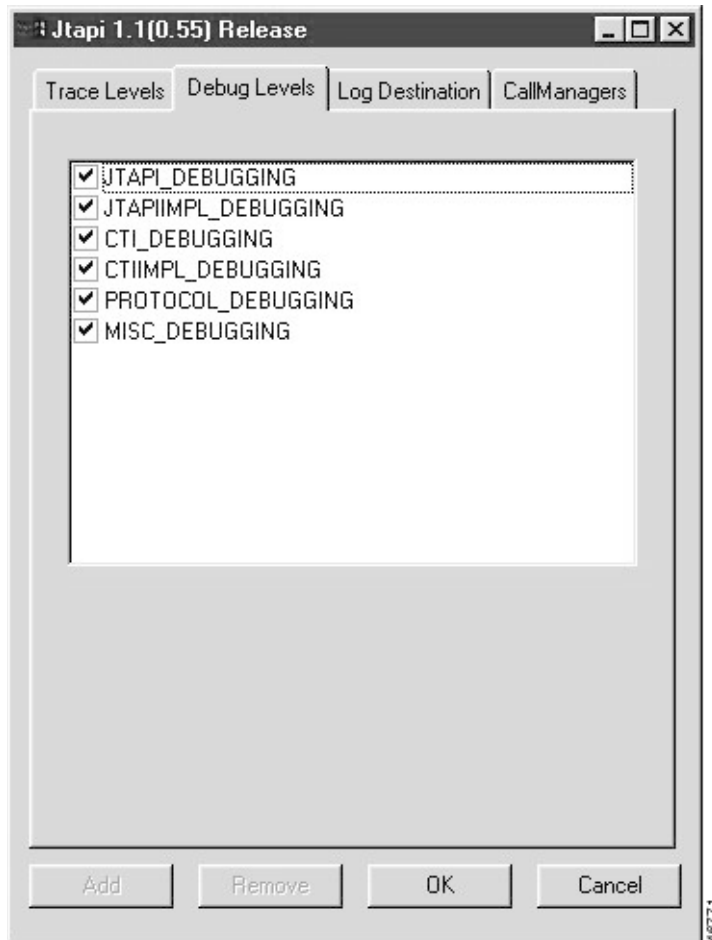
The Trace Level tab allows you to enable or disable the following JTAPl trace levels:

- WARNING - low level warning events
- INFORMATIONAL - status events
- DEBUG - highest level debugging events

Debug Levels

Figure 0-2 illustrates the debug level window.

Figure 0-2 Cisco JTAPI Debug Levels



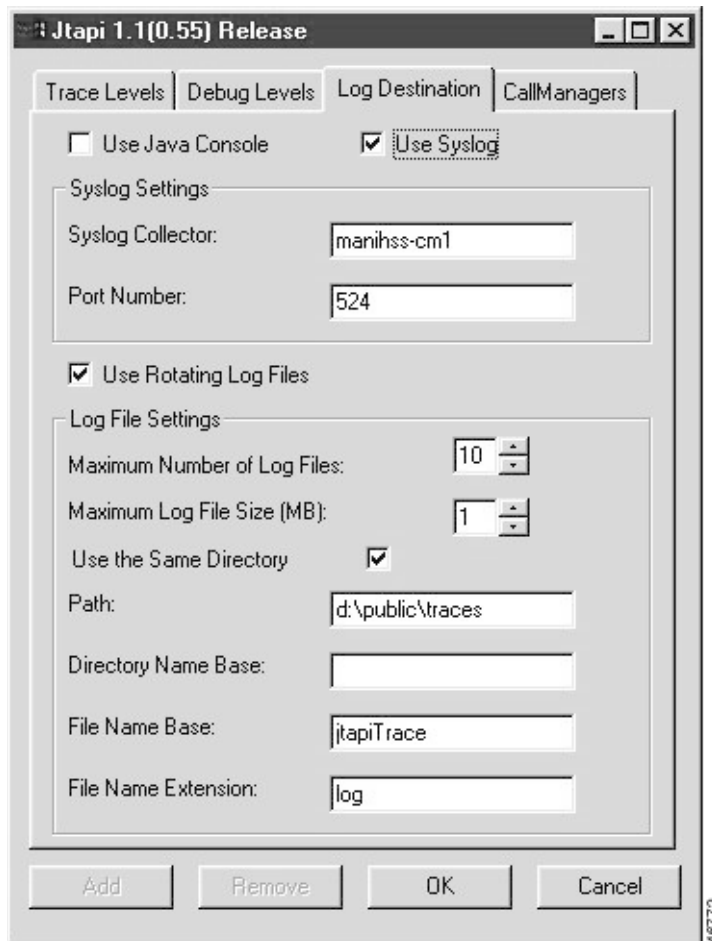
When the DEBUG trace level is enabled, you may enable or disable additional debugging levels in the Debug Levels tab, as described below:

- JTAPI_DEBUGGING - to trace JTAPI methods and events
- JTAPI_IMPLDEBUGGING - internal JTAPI implementation trace
- CTI_DEBUGGING - to trace Cisco CallManager events that are sent to the JTAPI implementation
- CTIIMPL_DEBUGGING - internal CTICLIENT implementation trace
- PROTOCOL_DEBUGGING - full CTI protocol decoding
- MISC_DEBUGGING - miscellaneous low-level debug trace

Log Destination

Figure 0-3 illustrates the log destination window.

Figure 0-3 Cisco JTAPI Log Destination



The Log Destination tab allows you to configure how JTAPI will create traces and how they will be stored. Log destination fields are described as follows:

Use Java Console

When this option is enabled, tracing is sent to the standard output or console (command) window.

Use Syslog

When this option is enabled, traces are sent to a UDP port as specified in the Syslog Collector and Port number fields. Syslog collector is the service by which traces are collected and directed to the CiscoWorks2000 server.

Use Rotating Log Files

This option allows you to direct the traces to a specific path and folder in the system. There can be no fewer than two log files and no more than ninety-nine. Cisco JTAPI rotates through the log files in numerical order, returning to the first log file after filling the last. Log files are sized in increments of one megabyte.

Use the Same Directory

This option allows you to specify if the same folder name should be used for each instance of an application. When enabled, JTAPI will trace the log files to the same directory. In this case, successive instances of a JTAPI application will restart the log files starting at index 01. When disabled, each application instance, whether successive or simultaneous, will cause the trace files to be placed in a new folder sequential to the last folder written. Cisco JTAPI detects the last folder present in the trace path and automatically increments the numeric index.

Path

Allows you to specify the path name where the trace files will be written to. When the path is not specified, the default is the application's path.

Directory Name Base

Allows you to specify a folder name where the trace files will be contained.

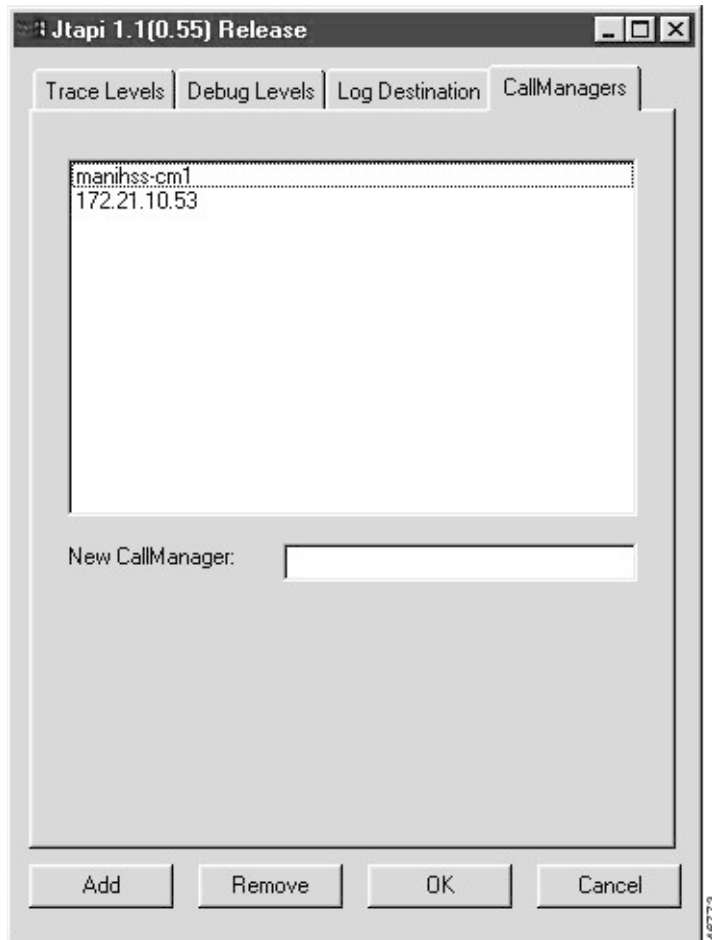
File Name Base and File Name Extension

These values are used to make up the trace file names with a numerical index appended to the file base name to indicate the order in which the files are created. For example, as shown in the figure the trace files would rotate between jtapiTrace01.log, jtapiTrace02.log and jtapiTrace10.log. If the 'File Name Base' and 'File Name Extension' were left blank Cisco JTAPI will pick the trace file names as CiscoJtapi01.log, CiscoJtapi02.log and so on.

Cisco CallManager

Figure 0-4 illustrates the Cisco CallManager window

Figure 0-4 Cisco JTAPI CallManager



This tab allows you to define a list of Cisco CallManagers that a JTAPI application can present the user for optional Cisco CallManager connectivity.

Other JTAPI Preferences Settings

There are a number of parameters that are not configurable through the JTAPI Preferences application. These parameters are low-level and may be needed only for troubleshooting and debugging purposes. Typically, you would not need to modify these unless instructed by support personnel. These parameters and their values must be edited manually in the jtapi.ini file located by default in %Sysroot%\java\lib. When editing parameters in the jtapi.ini file, use the format below:

```
<Parameter>=<value>
```

For example:

```
RouteSelectTimeout=5000
```

The following table lists the additional parameters, their functions, and default values.

Parameter	Function	Default
RouteSelectTimeout	Specifies the amount in milliseconds for JTAPI to wait for a response to a Route event	5000
PeriodicWakeupEnabled	Instructs the implementation to do periodic measurements of its internal queues	0
PeriodicWakeupInterval	Specifies the frequency in milliseconds to do periodic measurements	50
QueueStatsEnabled	Instructs the implementation to trace queue statistics	0
DeviceListUpdateEnabled	Instructs the implementation to update the list of provider controlled devices periodically	1
DeviceListUpdateInterval	Specifies the frequency in milliseconds to do device list updates	30

Parameter	Function	Default
AlwaysQueryCallInfo	Instructs the implementation to always query the call information for tracing purposes	1
CtiRequestTimeout	Specifies the time in milliseconds that JTAPI will wait for a response from a CTI request	30

JTAPI Preferences on non-Microsoft environments

For non-Microsoft environments, a jtapi.ini file must be created manually and placed in the CLASSPATH. The following are the parameter names with sample values:

```
Traces=WARNING; INFORMATIONAL; DEBUG
WARNING=0
INFORMATIONAL=0
DEBUG=1
Debugging=JTAPI_DEBUGGING; JTAPIIMPL_DEBUGGING; CTI_DEBUGGING;
          CTIIMPL_DEBUGGING; PROTOCOL_DEBUGGING; MISC_DEBUGGING
JTAPI_DEBUGGING=1
JTAPIIMPL_DEBUGGING=1
CTI_DEBUGGING=1
CTIIMPL_DEBUGGING=1
PROTOCOL_DEBUGGING=0
MISC_DEBUGGING=0
UseSystemDotOut=0
UseSyslog=0
SyslogCollector=
SyslogCollectorUDPPort=
UseTraceFile=1
TraceFileSize=1048576
NumTraceFiles=10
UseSameDirectory=1
Directory=
TracePath=d:/public/traces
FileNameBase=jtapiTrace
FileNameExtension=log
CallManagers=ctiserver; ctiserver2
```

Administering User Information for JTAPI and TAPI Applications

JTAPI and TAPI applications require that users be administered in the directory and be given privilege to control one or more devices. Follow the procedures for adding a user and assigning devices to a user in the “Adding a New User” section before using a JTAPI or TAPI application. The list of devices assigned to the user represent the phones that the user needs to control from the application (e.g. make calls, answer calls).