



LotusLive Meetings Options Kit

Silent or Automated Installations

This document provides the information you need to understand, plan, and deploy the IBM LotusLive Meetings, installation components in your environment.

This guide is intended for administrators who are responsible for performing silent or automated client component installations. It is written for organizations that plan to subscribe to the LotusLive Meetings service.

Please Note: The Outlook add-in (within the Options Kit) is for LotusLive Meetings only. LotusLive Events are scheduled via the service. LotusLive Events does not synchronize with the Outlook plug-ins. With LotusLive Events, all scheduling takes place from within the LotusLive Events scheduling pages.

THE INSTALLATION KIT

The Installation Kit is used to deliver sharing components to client PCs using a standard InstallShield kit. This installation typically requires user interaction to confirm various installation options while it is being executed. For large group installations it may be desirable to unbundle the installation components and deliver them to systems manually or to run the installation process in a silent, unattended mode.

To install:

- 1 Obtain the installation file for the appropriate brand. This Options Kit file can be downloaded from the login page at:
http://www.conferenceservers.com/brands/ITC/Wholesale/Wholesale_install.exe
- 2 Unpack the installation files to a folder using the following command line:
install.exe -noinst -r -f c:\wipInstallFiles
- 3 Once you have extracted the installation files to the folder you have specified (e.g.c:\ wipInstallFiles) you'll find a file called SETUP.EXE. The file SETUP.EXE is the main program file used for installing and it uses the following optional command line parameters, which must be preceded by a dash (-). When you use long path and file name expressions, enclose the expressions in double quotes.

Parameter	Description
-noinst	Use this option to cause setup to prompt you for installation choices and then to terminate without performing an actual installation. If you use -noinst it must be the first parameter after setup. Normally you will use -noinst with -r .
-r	Use this option to automatically record a silent installation response file (.ISS file) from an attended install. This is a record of the installation input and is stored, by default, in the windows directory.
-s	Use this option to run a silent installation.

- f1<path\ResponseFile>	Use this option to specify the alternate location and name of the response file (.ISS file). If you use this option when running InstallShield Silent , the response file is read from the directory or file specified by <path\ResponseFile>. The argument to -f1 currently cannot be longer than 43 characters. If your response file is a pathname longer than 43 characters, you can move the response file to a directory that results in a shorter pathname.
-f2<path\LogFile>	Use this option to specify the alternate location and name of the log file created by InstallShield Silent . By default, SETUP.LOG is created and stored in the same directory as SETUP.EXE. You must have write access to the location where SETUP.LOG will be created.

USAGE EXAMPLES FOR A SILENT INSTALL:

Command Line	Description
setup -s	Launches InstallShield Silent and loads SETUP.ISS from the directory containing SETUP.EXE. The log file SETUP.LOG is created in the same directory.
setup -s -f1c:\mydir\mydir.iss	Launches InstallShield Silent and uses MYDIR.ISS (from the C:\MYDIR directory) as the response file. This example also creates the log file SETUP.LOG in the same directory as SETUP.EXE .
setup -s -f1c:\mydir\mydir.iss -f2c:\mydir\mydir.log	Launches InstallShield Silent, uses MYDIR.ISS from the C:\MYDIR directory, and generates the log file MYDIR.LOG in the C:\MYDIR directory.
setup -noinst -r	Records a response file without installing the product. The purpose is to generate a response file for subsequent installations without doing an actual installation on the current system. By default, the response file is named SETUP.ISS and is located in the Windows directory.
setup -noinst -r -f1c:\mydir\mydir.iss	Records a response file without installing the product and generates a response file MYDIR.ISS in the C:\MYDIR directory

Usage Notes:

- 1 The parameters f1 and f2 cannot be followed by a space.
- 2 When you run InstallShield Silent, a log file is created in the directory where SETUP.EXE is located. By default, the log file has the default name SETUP.LOG if the -f2 parameter is not provided along with -f1.
- 3 If you don't use the -f1 parameter when running InstallShield Silent , the response file SETUP.ISS is assumed to be in the same directory as SETUP.EXE. A log file is created in the same directory.
4. SETUP.EXE command line parameters are not case-sensitive.

Troubleshooting

SERVER INSTALLATION LOGS

During a server installation two logs are created which show the files which have been transferred and errors encountered during the installation. These two files are named instlog.txt and instlogf.txt and are located in the Windows system directory which could be c:\windows or c:\winnt. As the installation is occurring various command shell windows will appear which may display errors which are being logged. Instlog.txt is used by the main installation (upgrade) process and instlogf.txt is used by the Full Server Installation (Full Install calls the Main Install).

APPLICATION VIEWER LOGGING

Application viewer logging is normally disabled, this can be enabled via the LoggingLevel field in the tbGateway table of the iGateway database. A value of 1 will disable logging, a value of 2 will enable it, other values are not supported. This will enable logging for all viewers connected to that database (it's generally useful to change this, start your viewer and then change it back). The log will be output in the Java Console of the client (note that you may need to enable the console first in the browser). Note that this will have an impact on performance, especially if the console is open, it is usually better to leave the console closed until the event you are trying to log has occurred

SERVER LOGGING

The following registry keys are created and assigned default values during a server installation. Each registry key is used for a different component but each shares the following common value assignments.

- + AUTOFLUSH: Boolean indicating if output to file should be flushed each time data is written (this is useful in cases of system or component crashes or hangs).
- + LOGPATH: location + name of the logfile, note that wiplogs directory is accessible via wadmin
- + MAXLENGTH: maximum length of the log in kbytes, when the length is reached the log is wrapped. (0x00001400 = 5 meg)
- + OVERRIDE: Boolean indicating if file should be created each time (1), or appended to (0)
- + TRACELEVEL: Specifies the types of activities to log. Multiple activities can be specified, i.e.. 0x003f would be all activities.

LevelError	= 0x0001
LevelWarning	= 0x0002
LevelInfo	= 0x0004
LevelFunctions	= 0x0008
LevelDebug	= 0x0010
LevelRepeatable	= 0x0020

LogLevel: verbosity of the logfiles, 0=off, 1-3 increasingly verbose, 4=very verbose, including dumps of data packets and other buffers. Note that only one of TRACELEVEL, or LogLevel will be used depending on the logging scheme used by the component.
[HKEY_LOCAL_MACHINE\SOFTWARE\WebDialogs\Collaboration\Tracing]

```
"AUTOFLUSH"=dword:00000000
"LOGPATH"="C:\wiplogs\csdll.log"
"MAXLENGTH"=dword:00001400
"OVERRIDE"=dword:00000000
"TRACELEVEL"=dword:00000007
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\WebDialogs\ConvertExe]
"LogPath"="C:\wiplogs\OffCnvt.log"
"LogLevel"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\WebDialogs\ConvertManager]
"LogLevel"=dword:00000000
"LogPath"="C:\wiplogs\ConvertManager.log"
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\WebDialogs\ds\Tracing]
"AUTOFLUSH"=dword:00000000
"LOGPATH"="C:\wiplogs\dsdll.log"
"MAXLENGTH"=dword:00001400
"OVERRIDE"=dword:00000000
"TRACELEVEL"=dword:00000007
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\WebDialogs\FileUploader\Tracing]
"LogPath"="C:\wiplogs\fdll.log"
"MaxPacketDump"=dword:00000000
"Tracelevel"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\WebDialogs\VoiceApi\Tracing]
"LOGPATH"="C:\wiplogs\voiceapi.log"
"TRACELEVEL"=dword:00000001
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\WebDialogs\WdApi\Tracing]
"LOGPATH"="C:\wiplogs\wdapi.log"
"TRACELEVEL"=dword:00000001
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\WebDialogs\WebAgent\Tracing]
"LOGPATH"="C:\wiplogs\webagent.log"
"TRACELEVEL"=dword:00000001
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\WebDialogs\XpUrl]
"LOGPATH"="C:\wiplogs\xp_url.log"
"TRACELEVEL"=dword:00000001
```

Enabling Client Logging

To enable logging on a client, modify the HKEY_LOCAL_MACHINE\SOFTWARE\WebDialogs\WDCapture key in the client machine's registry. The file path for the log file and amount of detail desired can be specified using the LogPath and LogLevel keys. Below are two registry files which can be used to enable or disable agent logging.

 [wdcapture.reg](#)

 [wdcaptureoff.reg](#)