

# MiCollab Advanced Messaging UCConnect System Administrator Guide

For version 9.1 and above

## Notice

The information contained in this document is believed to be accurate in all respects but is not warranted by Mitel Networks™ Corporation (MITEL®). Mitel makes no warranty of any kind with regards to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes.

No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from Mitel Networks Corporation.

## Trademarks

The trademarks, service marks, logos and graphics (collectively "Trademarks") appearing on Mitel's Internet sites or in its publications are registered and unregistered trademarks of Mitel Networks Corporation (MNC) or its subsidiaries (collectively "Mitel") or others. Use of the Trademarks is prohibited without the express consent from Mitel. Please contact our legal department at [legal@mitel.com](mailto:legal@mitel.com) for additional information. For a list of the worldwide Mitel Networks Corporation registered trademarks, please refer to the website: <http://www.mitel.com/trademarks>.

© Copyright 2020, Mitel Networks Corporation

All rights reserved

# Contents

<b>Preface</b>	<b>5</b>
References	5
Documentation	5
Documentation Updates	6
Help	6
Document Conventions	6
UCCONNECT Wizard VB.NET	8
UCCONNECT API Reference	8
Frequently Used Terms	8
<b>Overview</b>	<b>9</b>
Required Skills	9
<b>Installing UCCONNECT</b>	<b>10</b>
System Requirements	10
Call Server Platform	10
Developer Platform	11
Remote Platform	11
Voice and Fax Connections	12
Network Connections	12
<b>Installing UCCONNECT Software</b>	<b>13</b>
MiCollab AM Call Server Platforms	13
Developer Platform Option	13
Remote Platform Option	14
Installing UCCONNECT	14
Installing UCCONNECT on a Call Server	14
Installing UCCONNECT on a Developer or Remote Platform	16
<b>Starting the UCCONNECT Configuration Utility</b>	<b>18</b>
Viewing Line Status	19
<b>Configuring UCCONNECT</b>	<b>20</b>
Configuring UCCONNECT to Launch a Startup Script (Single Tenant System)	20
Configuring UCCONNECT to Launch Startup Scripts (Multi-Tenant System)	21

Configuring UConnect to Launch Scripts on a Local Machine	23
Editing the Node Exclusion List	23
<b>Testing the Installation</b>	<b>25</b>
Configuring a Call Processor Mailbox using the MiCollab AM Admin Utility	25
<b>Shutting Down MiCollab AM and the MiCollab AM UConnect Server</b>	<b>27</b>
<b>Developing UConnect Applications</b>	<b>28</b>
Visual Studio .NET Requirements	28
Using the Sample Applications	28
About AVST.UConnect.dll .NET	29
About SOAP Client API Access	29
Using the UConnect Application Wizard	29
Configuring the Script and Prompt File Location	30
<b>Sending Text Messages from UConnect Applications</b>	<b>32</b>
Compiling Applications	32
Recording Voice Prompts	32
Copying Script Prompts to Speech Directories	33
UConnect System Prompts	34
<b>Sending Faxes from UConnect Applications</b>	<b>35</b>
Creating and Sending Fax Documents	35
<b>Deploying Your Applications</b>	<b>37</b>
Running Scripts on a Remote Server Platform	37
Configuring the UConnect Service for Local and Remote Servers	38
<b>Setting Diagnostic Logging for UConnect</b>	<b>40</b>

# Preface

This guide describes how to install the UCCollab Interactive Voice Response (IVR) application.

This guide is written for Mitel-certified administrators and technicians who are familiar with MiCollab Advanced Messaging (MiCollab AM) procedures and terminology, and the Microsoft Windows® operating system.

Before implementing any procedures in this guide, ensure that MiCollab AM software is installed and running successfully.

To successfully implement UCCollab in an organization, the assistance of the following individuals, who constitute the implementation team, is required:

- MiCollab AM Administrator
- Telecom Technician
- MIS/IT Support Staff
- Software Developer

**IMPORTANT** Provide each member of the implementation team a copy of this guide several days or weeks before the implementation of UCCollab. This guide is best read sequentially to allow you to begin developing UCCollab IVR applications.

## References

A catalog of technical documentation is included on the MiCollab AM Installation Media. If you are installing any advanced applications, such as Networking and Fax Server applications, you should refer to the appropriate technical documentation for application and installation information.

## Documentation

The technical documentation is produced in the PDF format and requires the PDF reader to view it. The MiCollab AM Documentation Library includes the following documents and resources:

- **Administration Documentation.** Available as a PDF only. Contains the following:
  - **Administration Guides.** Available as a PDF only. Contains administrative guides for administrators about how to manage and configure the messaging system.
  - **Quick Reference Cards (QRC).** Contains shortcuts and quick instructions telling subscribers how to access and use the messaging system.
  - **User Guides.** Available as a PDF only. Contains user guides for subscribers about accessing the messaging system and checking and sending messages.
- **Server Documentation.** Available as a PDF only. Contains the following:

- **Developer Resources.** Contains programming guides and API references for developers for integrating the server clients and web applications with MiCollab AM.
- **Installation and Configuration.** Available as a PDF only. Contains installation and configuration guides for server administrators about how to install and configure the messaging system.
- **Integration Technical Notes (ITN).** Contains a set of guides that describe the integration methods and instructions for a variety of phone systems to work with MiCollab AM. The ITNs are generally used by resellers or administrators who are experienced with MiCollab AM and familiar with the integration procedures and terminology.
- **Spare Parts Documentation.** Contains a set of guides that describe the instructions for installing and configuring hardware parts to work with MiCollab AM. These documents are written for Mitel-certified MiCollab AM technicians who are experienced with MiCollab AM and familiar with the procedures and terminology.
- **Software Release Notice (SRN).** This notice introduces the new features, capabilities, and hardware/software requirements for the corresponding MiCollab AM version.

## Documentation Updates

Documentation updates may be available from the following sources:

- Mitel-certified technicians can view or download documents and program files from our partner web site: [www.mitel.com](http://www.mitel.com)

## Help

The primary source of information about MiCollab AM is the online help available within any of its administrative utilities. You can access **Help** by clicking the **Help** button in the dialog box or window in which you are working.

## Document Conventions

The following conventions are used in this document:

- **Key Names.** Names of keys on the keyboard are shown in a box.

Example: **Enter**

When two keys must be pressed simultaneously, they are joined by a + sign.

Example: **Alt** + **Tab**

- **Reference to Document** Titles of other documents are shown in italics.

Example: See the *System Installation and Configuration Guide*.

- **User Interface (UI) Element Names.** Names of UI elements such as dialog boxes, windows, screens, menu items, tabs, buttons, and icons are shown in bold.

**Example:** On the **Startup** screen, click the **Start** icon.

- **User Input.** Information required to be typed is shown in *italics*.

**Example:** Type the password *voicemail*.

- **Warning, Caution, Important, and Notes.** Text for the contents that require attention are shown as follows:

**WARNING** A warning paragraph advises you of circumstances that can result in the loss of data, harm to the MiCollab AM System Server platform, or personal harm.

**CAUTION** Failure to follow these recommendations can result in unauthorized access to the system and consequent loss of data.

**IMPORTANT** An important paragraph gives decision-making information or informs you of the order in which tasks need to be completed.

**NOTE** A note gives additional information, provides an explanation, or indicates an exception to the information in the preceding text.

For more related documents, refer to the following list of references:

Table 1. References

Document Number	Document Title
Administration Documentation	<i>System Administration Guide</i>
Server Documentation	<i>System Installation and Configuration Guide</i>
Server Documentation	<i>Call Processor Mailboxes Administration Guide</i>
Server Documentation	<i>XMediusFAX Integration Guide</i>
Server Documentation	<i>RightFax Integration Guide</i>
Server Documentation	<i>Faxtext for RightFax Integration Guide</i>
Online Help	<i>UCConnect Wizard VB.NET</i>
Online Help	<i>UCConnect API Reference</i>

## UConnect Wizard VB.NET

The primary source of information about UConnect Wizard is the online help available within the UConnect Wizard. You can consult UConnect Help by taking one of the following steps:

- Click the **Help** button in the UConnect Wizard.
- Press the **F1** key at any time.

## UConnect API Reference

Mitel provides the UConnect Help .NET API Reference with the UConnect application. To display the API Reference, from the Windows taskbar, go to **Start > All Program > MiCollab AM Desktop > UConnect > UConnect API Reference**.

## Frequently Used Terms

Table 2. Frequently Used Terms

Terms	Description
System Server	<p>Term refers to an organization's computer platform(s) that have MiCollab AM software installed and handles the core system functions such as storing messages, database.</p> <p>It can also refer generically to the System Server platform, the Call Server platform, or both. The term is most often used to describe a software or hardware installation or configuration practice where the role of the server platform is not specifically expressed.</p>
Call Server	<p>Term refers to an organization's computer platforms that have MiCollab AM software installed and serve as the interface to the system (PBX). The Call Server(s) interface with the System Server for the purpose of accessing messages, and database.</p>

# Overview

UCConnect is an Interactive Voice Response (IVR) component that allows you to build advanced computer telephony applications for the MiCollab AM system using any .NET language. With the vast array of third-party components available for .NET, you can develop custom IVR applications that use host access and database connectivity to provide sophisticated voice messaging, fax support, and other features to the enterprise.

## Required Skills

Though advanced expertise in telephony is not required to configure MiCollab AM for UCConnect, understanding MiCollab AM and telephony fundamentals is essential. If you are unfamiliar or uncomfortable with telephony concepts, consider using a certified MiCollab AM technician and a telecommunications technician to assist you.

Basic computer technician skills are required. If you are unfamiliar with installing peripherals on PCs, or configuring network connections, you may require assistance from a qualified PC technician. You should also have a basic understanding of the Microsoft Windows operating systems. If you are unfamiliar with terms such as Control Panel utilities and Windows services, you should have a knowledgeable technician assist you.

To develop custom IVR applications, seek out the services of a software developer with programming experience in a Microsoft .NET language. This typically requires a person with 1 to 2 years (or more) of experience writing .NET applications. If your application uses advanced database or host access or is particularly complex, you may require a programmer with experience in a particular area.

# Installing UConnect

You can install UConnect on the following three types of platforms:

- **Call Server platforms** – MiCollab AM Call Server platforms, on which most compiled UConnect scripts run, are production-level servers. Refer to the corresponding Software Release Notice for hardware and software requirements of specific platforms.

**NOTE** In this instance, Call Server platforms include a System Server with Call Services and Lines enabled.

- **Developer platforms** – A workstation or server platform on which a software developer creates and tests the IVR scripts for UConnect applications. Each developer platform must be equipped with a sound device, a microphone, and the Visual Studio development tool used to create the IVR scripts.
- **Remote servers** – A server platform on which compiled UConnect scripts may run. These are separate server platforms that communicate with the Call Server platform through the Local Area Network (LAN) connection. In addition, you must install UConnect on the Call Servers on which the scripts execute and with which the remote server communicates.

## System Requirements

This section discusses the system requirements and the required physical connections necessary to install UConnect on these platforms.

The following list specifies the requirements for each type of platform for UConnect. For specific information on the hardware and software requirements for MiCollab AM platforms, refer to the *Software Release Notice for MiCollab AM 9.1* or contact your MiCollab AM dealer. Review these requirements before continuing to the other sections in this guide.

### Call Server Platform

- Windows Server 2012 R2, Windows Server 2016 (Server with Desktop Experience), or Windows Server 2019 (Server with Desktop Experience)
- MiCollab AM 9.1
- UConnect version 9.1
- Mitel software feature license key enabled with UConnect
- Dual Core Intel® Celeron™ G3900 2.8 GHz CPU or better microprocessor
- 8 GB memory with or without speech
- Hard disk with at least 10GB of free space available on the Operating System Partition plus 10GB of free space on the Telephony Server partition
- Microsoft .NET 4.5.2

**NOTE** The UCCconnect installation installs the required .NET Framework automatically.

- Color VGA-compliant graphics adapter and monitor
- USB drive / DVD drive compatible with DVD+R media
- Windows-compatible network interface card (NIC) and connection to the LAN

## Developer Platform

- Windows 7, 8, 10, Microsoft Windows Server 2012 R2, Windows Server 2016 (Server with Desktop Experience), or Windows Server 2019 (Server with Desktop Experience)
- Microsoft Visual Studio 2012 or greater
- Microsoft .NET 4.5.2

**NOTE** The UCCconnect Developer installation installs the required .NET Framework automatically.

- Color VGA-compliant graphics adapter and monitor
- USB drive / DVD drive (compatible with DVD-R media)
- Sound card and microphone that support recording and playback of **.wav** files
- Sound editing software with the ability to do audio manipulations such as trimming silence at the beginning and end of a phrase

## Remote Platform

The UCCconnect remote server is a server platform dedicated to the execution of UCCconnect IVR scripts and connected to the MiCollab AM server through a network connection.

The following list represents the minimum hardware required to run UCCconnect IVR scripts on a remote platform. The processor and memory requirements for a specific remote UCCconnect platform depend on the size, complexity, and number of scripts the platform runs; the developer is responsible for determining the requirements necessary to run each script.

- Microsoft Windows Server 2012 R2, Windows Server 2016 (Server with Desktop Experience), or Windows Server 2019 (Server with Desktop Experience)
- Microsoft .NET 4.5.2

**NOTE** The UCCconnect Remote installation installs the required .NET Framework automatically.

- Color VGA-compliant display adapter and monitor
- A network interface card (NIC) and connection to the LAN
- USB drive / DVD drive compatible with DVD+R media

## Voice and Fax Connections

To run scripts in production mode, where callers use telephone lines to call into or out of the system, you must have a Call Server platform properly configured with voice linecard resources to support each voice line that serves MiCollab AM.

Optionally, to support a fax application, you must have either a functional XMediusFAX fax server or an OpenText RightFax fax server equipped with Brooktrout fax line cards. The XMediusFAX fax server or the RightFax fax server communicates to MiCollab AM servers and remote UConnect servers through the local area network connection.

For information about XMediusFAX, refer to the XMediusFAX documentation packaged with your XMedius software. For information about RightFax, refer to the RightFax documentation packaged with your RightFax software.

## Network Connections

MiCollab AM server platforms, remote UConnect platforms, and XMediusFAX fax server or RightFax fax server platforms require local area network connections in order to communicate with one another.

In addition, network connections allow system access for system administration, for system maintenance, and for network communication with the outside world.

Developer platforms may use network connections to upload scripts to MiCollab AM servers and remote servers. All MiCollab AM platforms require a network interface card regardless of whether they are connected to a LAN.

# Installing UConnect Software

You have several installation options for UConnect, and one or more of these options may be appropriate for your application. Please read this information carefully to determine which option(s) are right for you before attempting to install the software. This chapter discusses the tasks required to install UConnect on each platform type.

## MiCollab AM Call Server Platforms

The Call Server platform (includes the System Server with Call Services and Lines enabled) is the most commonly selected option. Use this option if you are configuring a system for an end-user site or for script development where you wish to run scripts in production mode using telephone lines to call into or out of the system.

The Call Server typically runs additional MiCollab AM application modules such as **Voice Mail**, **Automated Attendant**, **Automatic Speech Recognition**, or **Unified Messaging**.

You can use the Call Server platform to develop scripts and record prompts if a sound device, microphone, and a Microsoft .NET development tool is installed. However, Mitel recommends that you develop scripts and record prompts on developer platforms, and then you can copy the files to the Call Server. Developing scripts on production MiCollab AM systems may adversely affect call processing.

## Developer Platform Option

Install UConnect on a developer platform to create .NET scripts or record prompts on a different computer than the platforms on which they run, and then upload the scripts to MiCollab AM or a remote UConnect server platform.

Because voice ports are not required with this option, scripts run using a supplied Call Simulator utility that emulates a Touch-tone™ telephone and allows the developer to speak prompts through the platform's sound device as the script executes.

Software developers can create, test, and debug scripts on their workstations before uploading the script and allowing it to run on the Call Server platform. This alleviates runtime problems on a MiCollab AM system that is actively processing calls.

In addition, the developer platform requires .NET development tools to create, test, and debug scripts.

**IMPORTANT** You must also install any non-standard .NET assembly references used on the developer's platform on the Call Server that is executing the scripts.

## Remote Platform Option

The remote platform executes scripts on one or more remote servers that communicate with the Call Server platform through the network connection. Remote UCCconnect servers mitigate performance problems. Use a remote UCCconnect server for the following reasons:

- On systems that require a large number of ports or a large amount of CPU capacity
- On systems that have many simultaneous scripts executing
- On systems that have intensive database access requirements
- On systems where resource restrictions allow a limited number of scripts to run on a single platform

The remote platform offloads the intensive CPU and disk workload from the Call Server. You can dedicate scripts to run on the remote platform only, or split them to run on both the Call Server and the remote platforms.

The remote platform option is more complex to configure. You should deploy a remote server only when it is required. The remote server configuration significantly improves overall system performance when running some UCCconnect scripts; however, scripts that must start quickly or play and record large amounts of speech perform best on the Call Server platform itself.

## Installing UCCconnect

The following procedures provide instructions on how to install UCCconnect on MiCollab AM servers, developer platforms, and remote servers. There are two distinct installation processes:

- UCCconnect installs on MiCollab AM servers by installing the MiCollab AM Server component from the Server Components folder of the MiCollab AM 9.1 Installation Media, and then selecting UCCconnect from the list of software components during the setup process.
- UCCconnect installs on development and remote server platforms by installing the UCCconnect stand-alone component from the Developer Components folder of the MiCollab AM System Server 9.1.

## Installing UCCconnect on a Call Server

To install UCCconnect on a MiCollab AM platform, you must install the MiCollab AM server software. If MiCollab AM 9.1 is already running, you must shut down MiCollab AM before you begin, and then re-install MiCollab AM Server software.

### To install UCCconnect on a Call Server:

- 1 Log on to the platform using a Windows Administrator account.
- 2 Shut down all running programs.
- 3 Insert the MiCollab AM 9.1 Installation Media into the appropriate drive.
- 4 Perform one of the following:
  - If autorun is enabled, the MiCollab AM 9.1 Installation Media appears. In the Mitel MiCollab AM Installation Media **Components** area, click **MiCollab AM Server**, and then skip to **Step 6**.

- If autorun is not enabled, on the Windows taskbar, select **Start > Run > Browse**, and then continue to **Step 5**.

- 5 Locate and open the ...\\Server Installs\\Telephony Server folder, double-click **start.hta**, and then click **OK**. The MiCollab AM Installation main window appears.
- 6 In the **Server Components** section, select **MiCollab AM Server**. The **Welcome** page appears.
- 7 Click **Next**. The **License Agreement** appears.
- 8 Click **Yes** to accept the License Agreement.

If you have not installed the **MiCollab AM Speech and TTS Support** components, a warning appears.

**IMPORTANT** You must accept the terms of the license agreement to continue with Setup.

- 9 Click **Next** to continue with the next step in the installation. The **Select Hardware Support Components** page appears.
- 10 Select the checkboxes in the **Software** list to choose the following individual components. Select only the components you require for the installation.

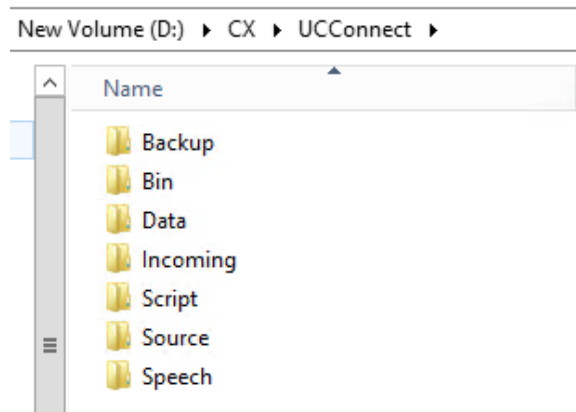
**NOTE** If installation detects the current version of any of the software packages, the corresponding checkbox is not available.

- 11 Click **Next** to continue. The **Select Components** page appears.
- 12 In the **Components** list, select **UConnect**.

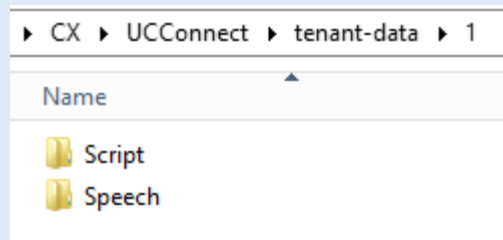
Verify that the **Destination Folder** points to the location where you want MiCollab AM installed. If you want to change the default destination folder, click **Browse** and then type or select the drive and folder destination from the list.

**IMPORTANT** Mitel recommends that you install MiCollab AM software on a partition separate from the operating system. By default, MiCollab AM software installs on the **D:** drive partition in the **MiCollab AM** folder.

- 13 Click **Next**. The **Start Copying Files** page appears.
- 14 Setup lists the components currently selected for installation. Verify the components you are about to install, and then click **Next**.
- 15 The setup wizard continues through the first part of the installation, and then the **System Restart** page appears.
- 16 Remove all disks from the computer's drives, and then click **Finish** to restart your system.
- 17 Log on as the administrator after the platform restarts. A pop-up dialog box appears, alerting you that the installation is resuming, and then the installation completes.
- 18 The **UConnect** folder appears in the installation directory. You can expand this folder to see the application support directories.



**NOTE** If you are licensed for multi-tenanting, you need to create a separate **tenant-data** folder and a **{tenant-id}** folder (where {tenant-id} is the tenant id number) for each tenant and then copy and paste the **Script** and **Speech** folders into the new {tenant-id} folders. This enables the applications to be used by each tenant.



You can find the tenant-ids by clicking the **Tenant Info** button on the **UCCconnect Configuration** utility. This utility can be found in the Windows Control Panel.

## Installing UCCconnect on a Developer or Remote Platform

To install UCCconnect on a developer or remote platform:

- 1 Log on to the platform using a Windows Administrator account.
- 2 Shut down all running programs.
- 3 Insert the MiCollab AM 9.1 Installation Media into the appropriate drive.
- 4 Perform one of the following:
  - If autorun is enabled, the MiCollab AM 9.1 Installation Media appears. In the Mitel MiCollab AM Installation Media **Components** area, click **MiCollab AM Server**, and then skip to **Step 6**.
  - If autorun is not enabled, on the Windows taskbar, select **Start > Run > Browse**, and then continue to **Step 5**.
- 5 Locate and open the ...\\Server Installs\\Telephony Server folder, double-click **start.hta**, and then click **OK**. The MiCollab AM Installation main window appears.

- 6 In the **Developer Components** section, select **UCConnect (stand alone)**. The **Welcome** page appears.
- 7 Click **Next**. The **License Agreement** appears.
- 8 Click **Yes** to accept the License Agreement.

**IMPORTANT** You must accept the terms of the license agreement to continue with Setup.

- 9 Click **Next** to continue with the next step in the installation. The **Select Components** dialog box appears.
- 10 Perform one of the following:
  - If you are installing UCConnect on a Developer platform,
    - Select **UCConnect Developer**.
    - Select **SNMP Support** if you want to install SNMP support for UCConnect on the Developer platform.
  - If you are installing UCConnect on a Remote platform,
    - Select **UCConnect Remote**.
- 11 Accept the **Destination** folder or click **Browse** to select another location.
- 12 Click **Next**. The **Start Copying Files** dialog box appears.
- 13 Review the current installation path, and then click **Next**.
- 14 Once the software is installed the **Dr. Watson for Windows Configuration** dialog box appears. Click **Yes** to enable logging to the MiCollab AM log files, or click **No** to leave the settings at default.
- 15 Click **Finish** to complete the installation process and restart the computer.
- 16 Log on as the administrator after the platform restarts. A pop-up dialog box appears, alerting you that the installation is resuming, and then the installation completes.
- 17 The **UCConnect** folder appears in the installation directory. You can expand this folder to see the application support directories.

# Starting the UConnect Configuration Utility

After installing UConnect, you can then access the **MiCollab AM Configuration** utility and the **UConnect Configuration** utility from within the Windows Control Panel.

The **MiCollab AM Configuration** utility provides general configuration settings for MiCollab AM and provides the ability to start and stop the Call Server. For more information on the **MiCollab AM Configuration** utility, see the *System Administration Guide*, *System Installation and Configuration Guide*, and the online help.

The **UConnect Configuration** utility is added to the Windows Control Panel as part of the UConnect installation when you install UConnect on a MiCollab AM Call Server, a developer platform, or a remote platform. The **UConnect Configuration** utility provides configuration settings for UConnect and provides the ability to start and stop the MiCollab AM UConnect Server.

**NOTE** To run UConnect on a Call Server, the MiCollab AM service must be running. Any attempt to start the MiCollab AM UConnect Service will force an automatic start of the MiCollab AM service.

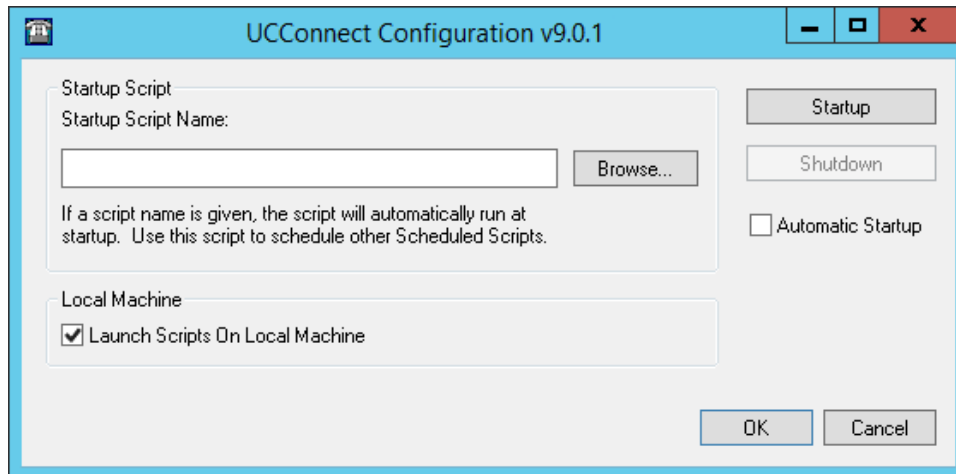
## To start MiCollab AM:

- 1 Open **MiCollab AM Configuration** and select the **Main** tab.
- 2 Click **Startup** and wait until **Current Status** displays **Running**.

## To start the MiCollab AM UConnect Server:

**NOTE** To run UConnect on a Call Server, the MiCollab AM service must be running.

- 1 Select **Start > Control Panel > Programs**, and then double-click **UConnect Configuration**. The **UConnect Configuration** utility appears.



- 2 Click **Startup**.

**IMPORTANT** If you are doing one or more remote installations, you must start up the MiCollab AM UCCONNECT Server on each remote platform.

- 3 Select the **Automatic Startup** check box to automatically start the MiCollab AM UCCONNECT Server at startup after you have successfully tested both the **MiCollab AM Configuration** and **UCCONNECT Configuration** utilities. Refer to [Testing the Installation](#).

**NOTE** Both the **MiCollab AM Configuration** and **UCCONNECT Configuration** utilities allow for automatic startup of each service when the platform starts. These utilities run as Windows services, allowing them to start automatically when the operating system starts.

However, when they are first installed, these two services default to manual startup initially, which allows you to configure and test them properly before they run. Testing includes calling in and successfully executing a sample script. Once you have finished installing and testing the **Call Server** and **UCCONNECT** software, be sure to set both Windows Control Panel utilities for automatic startup.

## Viewing Line Status

You can determine the status of each line on the Call Server using the **Line Status** utility. When a UCCONNECT call is being processed, the **Line Status** displays **UC <name>** where <name> is the name of the script that is executing. For more information on the **Line Status** utility, see the MiCollab AM help topic.

# Configuring UConnect

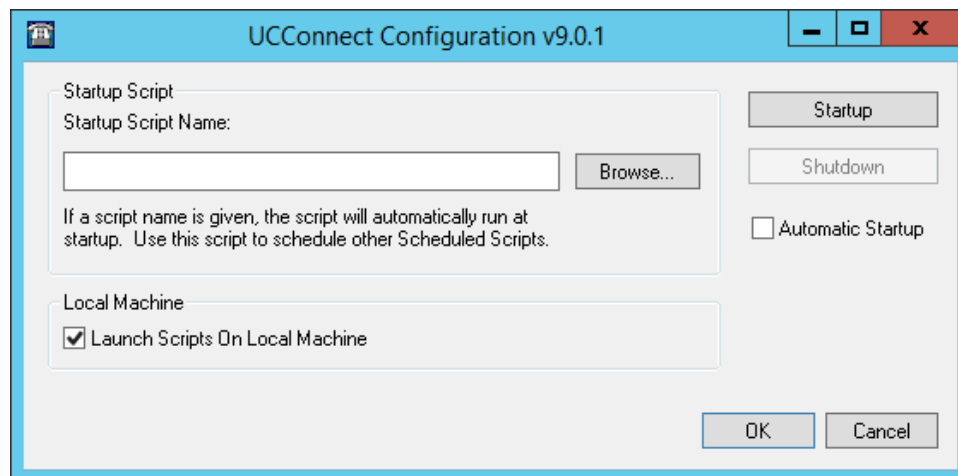
## Configuring UConnect to Launch a Startup Script (Single Tenant System)

For a system that is licensed for a single tenant, the **UConnect Configuration** utility enables you to launch a startup script automatically when the UConnect service starts. The startup script will run one time shortly after the service is started.

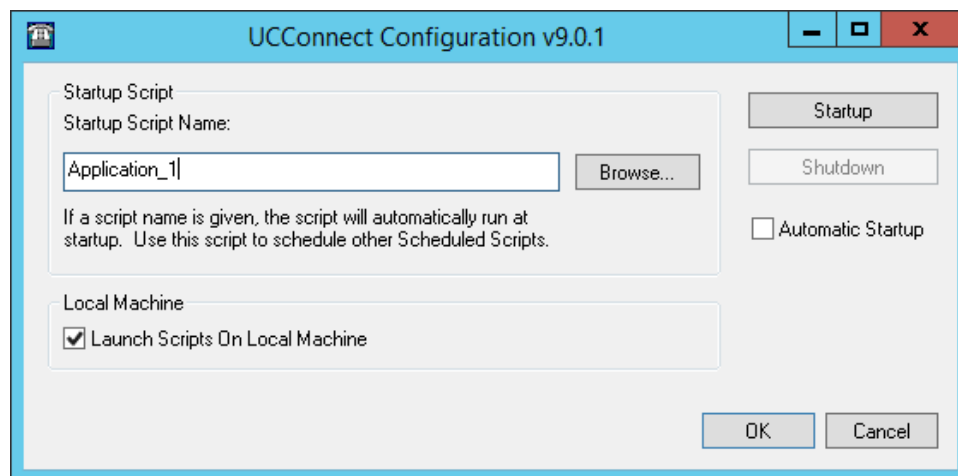
**For example:** Use startup scripts for outbound calling or to log on to host computers.

To select a startup script:

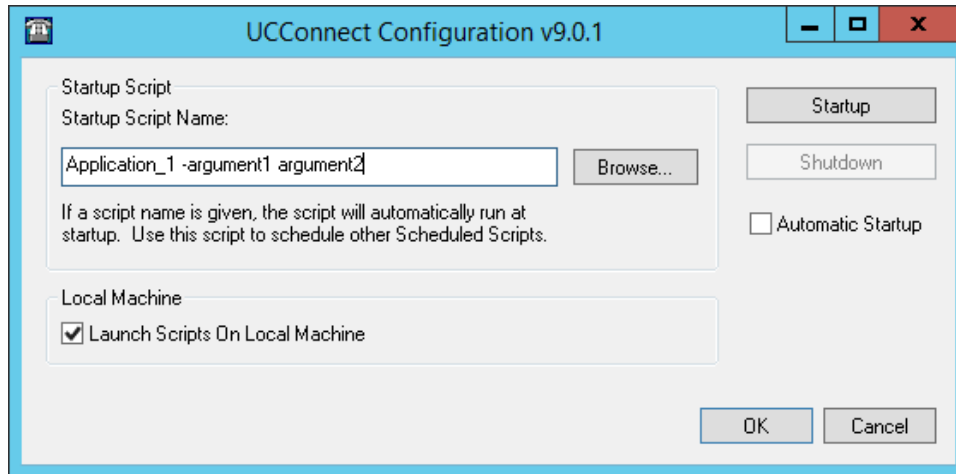
- 1 In the **Startup Script** area of the **UConnect Configuration** utility, click the **Browse** button to select a startup script to automatically run at startup. You can use this script to schedule other Scheduled Scripts.



- 2 Select a script file, and then click **Open**. The script name appears in the **Startup Script Name** text box.



- (Optional) Add command line arguments to the **Startup Script Name** as you normally would when executing any application. All data after the script name is provided as arguments to the command line when launching the startup script. These arguments can be in any format with or without "-", "/", etc.



- Click **OK**.

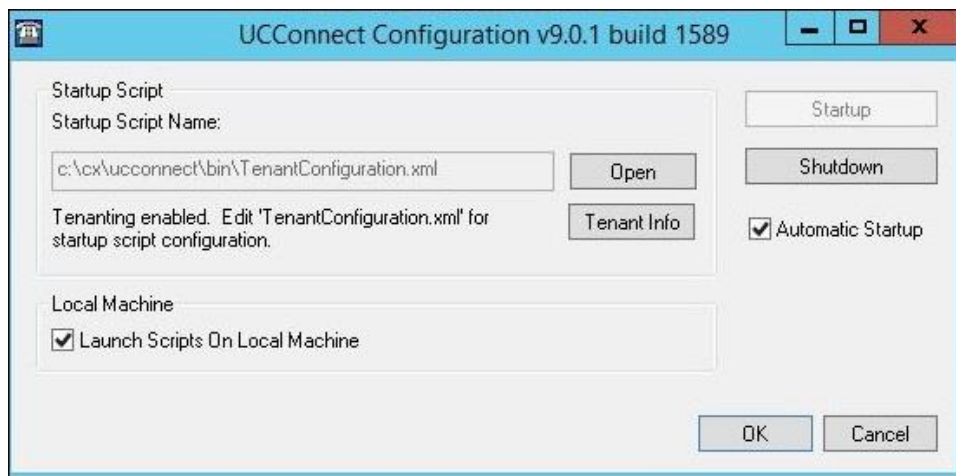
## Configuring UCCONNECT to Launch Startup Scripts (Multi-Tenant System)

For a system that is licensed for multi-tenanting, the **UCCONNECT Configuration** utility enables you to launch startup scripts for each tenant and edit the configuration file for each tenant. The path to the configuration file for the tenant (TenantConfiguration.xml) appears in the **Startup Script Name** box.

The TenantConfiguration.xml configuration file is a system-wide configuration that supports all tenants, and is not specific to each tenant.

### To edit a startup script for a tenant:

- In the **UCCONNECT Configuration** utility, in the **Startup Script** area, click the **Open** button.

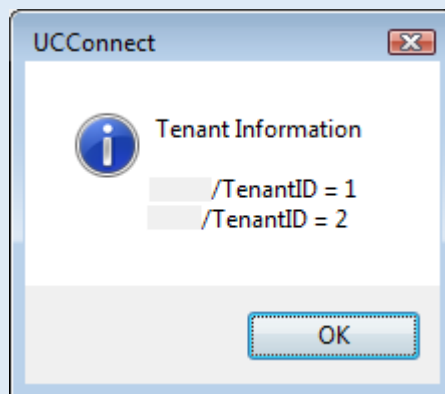


- 2 The TenantConfiguration.xml configuration file opens.
- 3 Define the command-line arguments for each startup application.

For example:

```
<install-dir>UCConnect/Bin/TenantConfiguration.xml
<TenantConfiguration>
  <ScriptStartupList>
    <!--
      Application_1 will be launched from"
    /uconnect/tenant-data/1/script/Application_1.exe
    -->
    <ScriptStartup>
      <TenantID>1</TenantID>
      <Name>Application_1</Name>
      <Arguments></Arguments>
    </ScriptStartup>
    <!--
      Application_1 will be launched from:
    /uconnect/tenant-data/2/script/Application_1.exe
    -->
    <ScriptStartup>
      <TenantID>2</TenantID>
      <Name>Application_1</Name>
      <Arguments></Arguments>
    </ScriptStartup>
  </ScriptStartupList>
</TenantConfiguration>
```

**NOTE** You can click the **Tenant Info** button to see the tenant-id for each tenant.



- 4 Save and close the TenantConfiguration.xml file.

# Configuring UConnect to Launch Scripts on a Local Machine

Follow the guidelines listed below, depending on the type of platform installation.

- For Call Server platform installations: select the **Launch Scripts on Local Machine** checkbox.
- For Remote platform installations:
  - If you intend to open scripts from multiple machines – that is, if your scripts run on remote servers – you may choose to share the running of scripts between the Call Server platform and the remote machines, select the **Launch Scripts on Local Machine** checkbox.
  - If you want to force the scripts to run only on the remote machines, clear the **Launch Scripts on Local Machine** check box.
- For Developer platform installations: select the **Launch Scripts on Local Machine** check box.

## Editing the Node Exclusion List

The Node Exclusion List (NodeExclusionList.xml) is a configuration file that provides a list of Call Server nodes that should be excluded when launching a particular application. The NodeExclusionList.xml configuration file is located in ...\\UConnect\\Bin and has the following structure:

**NOTE** The NodeExclusionList.xml configuration file only applies to “outbound” calling scripts.

```
<AutomatedAgent>
  <AutomatedAgentApplicationData>
    <AutomatedAgentApplicationName>Application_1</AutomatedAgentApplicationName>
    <NodeExclusionList>
      <CXNodeID>100</CXNodeID>
      <CXNodeID>109</CXNodeID>
    </NodeExclusionList>
  </AutomatedAgentApplicationData>
  <AutomatedAgentApplicationData>
    <AutomatedAgentApplicationName>Application_2</AutomatedAgentApplicationName>
    <NodeExclusionList>
      <CXNodeID>202</CXNodeID>
      <CXNodeID>210</CXNodeID>
    </NodeExclusionList>
  </AutomatedAgentApplicationData>
</AutomatedAgent>
```

For systems licensed for multi-tenanting, a TenantID needs to be manually added to this configuration file in order for UCConnect to run. You can click the **Tenant Info** button on the **UCConnect Configuration** utility to see the TenantID for each tenant.

For example:

```
<AutomatedAgent>
  <AutomatedAgentApplicationData>
    <TenantID>1</TenantID>
    <AutomatedAgentApplicationName>Application_1</AutomatedAgentApplicationName>
    <NodeExclusionList>
      <CXNodeID>100</CXNodeID>
      <CXNodeID>109</CXNodeID>
    </NodeExclusionList>
  </AutomatedAgentApplicationData>
  <AutomatedAgentApplicationData>
    <TenantID>1</TenantID>
    <AutomatedAgentApplicationName>Application_2</AutomatedAgentApplicationName>
    <NodeExclusionList>
      <CXNodeID>202</CXNodeID>
      <CXNodeID>210</CXNodeID>
    </NodeExclusionList>
  </AutomatedAgentApplicationData>
</AutomatedAgent>
```

# Testing the Installation

To test UCCconnect after you have installed it, you can configure a Call Processor mailbox on the Call Server to run the **GUESS** sample script. **GUESS** is a simple guessing game in which the script generates a random number between 1 and 100 and gives the caller several chances to guess the correct number.

## Configuring a Call Processor Mailbox using the MiCollab AM Admin Utility

**MiCollab AM Admin** allows you to configure UCCconnect to run scripts for inbound calls from Call Processor mailboxes. Typically, MiCollab AM offers the caller a list of options from the answer mode Call Processor mailbox, the automated attendant. Callers can initiate UCCconnect scripts from the answer mode Call Processor mailbox or from any other Call Processor mailbox configured to run scripts.

For more information about **MiCollab AM Admin** or Call Processor mailboxes, refer to the online help system, *System Administration Guide*, and *Call Processor Administration Guide*.

You can initiate scripts from a Call Processor mailbox using **Speech** commands, by entering digits from a Touch-tone telephone, or through a time-out action of the Call Processor mailbox. Script names must be seventeen characters or less. Do not include the **.exe** file extension.

### To configure a Call Processor mailbox to run GUESS:

- 1 Make sure that MiCollab AM is running.
- 2 Log on to **MiCollab AM Admin** from the System Server or a client workstation.
- 3 Select a Call Processor mailbox or create one if necessary.
- 4 In the **Call Processor Actions** section, select the **View** dropdown list, and then select **DTMF Only**.
- 5 Select an unused DTMF Key to use for the test.
- 6 In the **Action** field of the selected key, select **Open Script**.
- 7 In the **Arguments** field, type "Guess".

**NOTE** You must enclose the script in quotes.

- 8 Click **OK**.
- 9 Perform a test call.
  - a Place a call to MiCollab AM, log on to the Call Processor if necessary.
  - b Dial the digit you configured for the test.
  - c The **GUESS** script begins.

- d** If you hear the prompt, *"Pick a number between 1 and 100,"* you are running the **GUESS** script.

# Shutting Down MiCollab AM and the MiCollab AM UConnect Server

Shut down MiCollab AM and the MiCollab AM UConnect Server only during periods of low call volume. During a shutdown, the system waits until all callers have disconnected before completing the process, and no new callers can access the system. It may take several minutes to complete the shutdown process, depending on the configuration.

## To shut down MiCollab AM and the MiCollab AM UConnect Server:

**IMPORTANT** Shutting down MiCollab AM will automatically shut down the MiCollab AM UConnect Server.

If you shut down the MiCollab AM UConnect Server before you shut down MiCollab AM, MiCollab AM will not shut down automatically. Callers could process a request to initiate a script; however, if the MiCollab AM UConnect Server is shut down, the script will not run.

- 1 Open **MiCollab AM Configuration** and select the **Main** tab.
- 2 Click **Shutdown** and wait until **Current Status** displays **Stopped**.
- 3 Open the **UConnect Configuration** utility and note that the **Shutdown** button is greyed out. This indicates that the UConnect Server is shut down.

**IMPORTANT** If you are using UConnect Remote Services, the UConnect Remote Services must be manually restarted, because they do not re-establish communications with the Call Server automatically.

# Developing UConnect Applications

You can use any of the .NET languages (VB, C# or C++) to develop debug your own scripts. However, if you use any of the UConnect sample scripts as a starting point, you must use VB.NET to modify the script.

If you use the Visual Studio .NET development environment, you can step line-by-line through the script execution, set breakpoints, create watch points, and monitor the flow of execution just as you would when developing and debugging any .NET application.

Use the call simulator on a developer system to input Touch-tone and play prompts over a sound device. On a Call Server platform, you can call into MiCollab AM and run the script or use the call simulator, if you have a sound device installed.

**NOTE** To develop and debug scripts directly on the call server platform you must have Visual Studio .NET installed on the platform, regardless of whether you use the call simulator or call in to run the script.

Developing scripts from the Call Server platform is not a recommended practice and may cause call processing problems on a working system.

## Visual Studio .NET Requirements

- The included Sample Applications and any Applications that are created using the UConnect Application Wizard require Visual Studio .NET 2015 or greater.

**NOTE** If using a version of Visual Studio greater than 2015, the sample projects will be automatically upgraded by the newer version.

- The UConnect .NET Assembly is built using .NET Framework 4.5.2. If you are creating new Projects, Visual Studio .NET 2012 or greater is supported.

## Using the Sample Applications

UConnect provides a number of sample application scripts to help you get started. These scripts provide examples of the various types of VB.NET scripts and show ways to use the various IVR programming functions.

Studying the samples is a quick and effective way of learning to develop UConnect scripts, and you can use their code fragments as resources once you start developing your own applications.

As a starting point, start the VB.NET development environment and load the **GUESS** sample script. If you installed UConnect in the default path, you can locate this application in the **...\CX\UConnect\Source\Samples\Guess** folder.

Next, examine the VB.NET code for this application. The application is initiated within the **Public Sub Main()** procedure. This is standard for most UCCConnect applications.

**NOTE** Refer to the AVST.UCCConnect.chm help file for the full API reference.

## About AVST.UCCConnect.dll .NET

UCCConnect uses a .NET DLL Assembly module (**AVST.UCCConnect.DLL**) to provide methods for implementing IVR functions.

**For example:**

There are methods to collect Touch-tone digits, speak prompts, and transfer calls.

**AVST.UCCConnect.DLL** installs and is automatically registered during the UCCConnect software installation.

Each of the sample application projects references the **AVST.UCCConnect.DLL**. If you are creating a new project, you must first create a reference (import) of the **AVST.UCCConnect.DLL**. This file is located in the ...\**CX\UCCConnect\Bin** folder.

The **Project Properties References** tab displays the location of the **AVST.UCCConnect.dll** file.

## About SOAP Client API Access

UCCConnect provides direct access to the telephony system SOAP Server within the UCCConnect Development Environment.

For more information on using this API, refer to the help file, **UCCConnect API Reference.chm**, which is located in the start menu of the **MiCollab AM Desktop** group on systems with UCCConnect installed.

For more information on the telephony system SOAP API, refer to the *MiCollab AM Web Services API Developer's Guide* located on the telephony system Installation Media.

## Using the UCCConnect Application Wizard

UCCConnect provides a utility called the UCCConnect Wizard – VB.NET to simplify the process of creating new application scripts. The use of the UCCConnect Wizard to create scripts is not a requirement, but recommended, in order to save time in both the development and the debugging of your script.

By default, the UCCConnect Wizard simply generates a script framework using a file name that you provide. This framework provides initialization, exit, and default error handling routines that enable you to begin designing your application immediately. You can also choose to add up to four predefined routines or code segments to the framework before generating the script and exiting UCCConnect Wizard.

To start the UCCConnect Application Wizard, select **Start > Programs > MiCollab AM Desktop > UCCConnect > UCCConnect Wizard - VB.NET**.

**NOTE** At any time while using the Wizard, click **Help** or press **F1** to get specific information about the utility.

The available code segments include:

- **Create a menu of options** – allows you to define:
  - The keys you want in your menu
  - A voice prompt to describe the menu to a caller
  - Additional prompts to confirm each key press in the menu
  - How to respond to an invalid selection or no selection at all
  - Prompt for input from the caller—allows you to define:
    - A prompt to request a string of digits from a caller
    - Specify how the string should be validated
  - Set responses for invalid input or no input at all
- **Take a message** – allows you to specify:
  - The Subscriber mailbox for collecting voice messages
  - The Subscriber mailbox used as the source, and optionally define a maximum length for each message
  - Transfer call to an extension—allows you to define:
    - A destination device number for incoming calls
  - The type of transfer you want the script to use

## Configuring the Script and Prompt File Location

If you create a script manually, the first thing you should do is configure the script so that it knows where to look for its speech prompts. To do this, you must set the **ScriptSpeechDir** property to the name of the directory that contains the prompts.

**For example:**

```
UCC.ScriptSpeechDir = "FirstApp"
```

**NOTE** The prompt directory that you specify in the **ScriptSpeechDir** property must be located in a sub-directory of the **...\\CX\\UCCconnect\\Speech** directory.

The property specifies a relative path, not an absolute one. For example, if the **ScriptSpeechDir** is set to **FirstApp**, the script attempts to retrieve its prompts from the **...\\CX\\UCCconnect\\Speech\\FirstApp** directory.

In most cases, you should create a separate directory for each script you develop. If you use the UCCconnect Wizard to create your script, the Wizard creates this directory and sets the **ScriptSpeechDir** property in your script automatically. By convention, the directory has the same name as the script. However, if you want two scripts to share a single prompt set, you can set the **ScriptSpeechDir** properties of both scripts to the same directory name.

**NOTE** If you write two scripts to share one prompt directory, you must also design their installation scripts to install their prompts in the shared prompt directory. For more information, refer to [Deploying Your Applications](#) later in this document.

# Sending Text Messages from UConnect Applications

UConnect includes a method for sending SMS or Text Messages directly from the UConnect Development Environment.

For more details, please refer to the help file, **UConnect API Reference.chm**, which is located in the start menu of the **MiCollab AM Desktop** group on systems with UConnect installed.

## Compiling Applications

Before running or debugging a script, you must compile it into an executable (.exe file) program.

To install a new version of a script while a previous version of the script is running, copy the new executable into the `...\CX\UConnect\Incoming\Script` folder. UConnect then transfers the executable to the `...\CX\UConnect\Script` directory, where it runs "production" applications on the Call Server platform. The new version runs automatically the next time the script initiates.

UConnect creates and runs a separate temporary version of the executable file on each MiCollab AM line from the `...\CX\UConnect\Script` directory. This makes it possible to install a new version of a script while a previous version of the script is running.

**NOTE** For a system licensed for multi-tenanting, this executable file will be found in the `...\CX\UConnect\tenant-data\{tenant-id}\Script` folder.

**NOTE** When compiling applications in Visual Basic, in order to use the Call Simulator to test the included Sample Applications, or applications created with the UConnect Wizard, you must insert **-d** in the **Command line arguments** field of the **Debug** tab in the Visual Basic compiler.

## Recording Voice Prompts

You must record and save UConnect prompts as **.wav** files in the following audio format:

- Linear PCM encoding, G.711 A-law, or G.711 u-law
- 8.0 KHz (8,000 samples per second)
- 8 bits per sample
- Monaural
- No data compression applied

**NOTE** The G.711 u-law recording format is typically used in the United States and Japan. The G.711 A-law recording format is typically used in South America and Europe.

If your IVR application requires complex prompts, large numbers of drop-in elements such as letters and numbers, or effects such as music under the spoken text, consider purchasing and using an audio waveform editor. These programs are available at a variety of prices and provide different levels of editing and special effect features.

## Copying Script Prompts to Speech Directories

By default, the Speech directory name is the same as the application name. Prompts for UConnect scripts are stored in directories (folders) under the `...\CX\UConnect\Speech` directory.

**NOTE** For a system licensed for multi-tenanting, the speech prompts are found in the `...\CX\UConnect\tenant-data\{tenant-id}\Speech` folder.

There is generally a separate directory for each script, and by convention, the directory shares the same name as the script.

### For example:

If you had a script named **Lotto**, the script's speech prompts would typically be stored in `...\CX\UConnect\Speech\Lotto`.

For a system licensed for multi-tenanting, the script's speech prompts would be found in the `...\CX\UConnect\tenant-data\{tenant-id}\Speech\Lotto` folder.

As previously stated, the **ScriptSpeechDir** property of the UConnect object module must be set to the relative path where the prompts are stored. In this example, the **ScriptSpeechDir** property is **Lotto**.

A unique feature of UConnect is that you can record new prompts on a system that is running and processing calls. To utilize this feature, you must record prompts into a **temporary** directory and then move them into the appropriate subdirectory below the `...\CX\UConnect\Incoming\Speech` directory. This directory structure is set up to mirror the `...\CX\UConnect\Speech` directory.

**NOTE** For a system licensed for multi-tenanting, the incoming speech subdirectories are found in the `...\CX\UConnect\tenant-data\{tenant-id}\Incoming\Speech` folder.

When you copy a prompt to a directory under `...\CX\UConnect\Incoming\Speech`, it moves automatically to the mirror location of `...\CX\UConnect\Speech` as soon as the prompt is not in use by the system.

If the prompt already exists in the destination directory, the new prompt replaces it. Using the Lotto example, if you copy a new prompt into the `...\CX\UConnect\Incoming\Speech\Lotto` directory, it moves automatically into the `...\CX\UConnect\Speech\Lotto` directory.

Recording or copying prompts directly to directories below the `...\CX\UConnect\Speech` directory while the UConnect service is running is not an acceptable practice, if there is a chance that a caller may be listening to the prompt at the same time. If there are no callers on the system while you are recording prompts, this is not a problem.

## UCConnect System Prompts

UCConnect system prompts are stored in a subdirectory of `...\CX\UCConnect\Speech`.

**NOTE** For a system licensed for multi-tenanting, the speech prompts are found in the `...\CX\UCConnect\tenant-data\{tenant-id}\Speech` folder.

The **SystemSpeechDir** property of the UCConnect control sets the name of this subdirectory for each script, but that name is usually System. If you are going to use a custom prompt set, Mitel recommends that you create a custom system speech directory and then set the **SystemSpeechDir** property to the new location instead of replacing the system prompts directly.

To retrieve and play a prompt, UCConnect searches the speech directory belonging to the active script first. If it cannot find the prompt there, it searches the system speech directory. This allows you to record only the specialized prompts that your script needs and retrieve standard prompts, such as numbers and months, from the system directory.

It also enables you to override any specific system prompt by recording a prompt of the same name in your script's speech directory. You may also re-record all of the prompts in the **System** directory so that your application uses a single voice. Take extra care when editing system prompts so that they concatenate smoothly when building numbers, dates, and other phrases.

# Sending Faxes from UConnect Applications

You can send faxes from within IVR scripts if you integrate either an XMediusFAX fax server or an OpenText RightFax Enterprise fax server with MiCollab AM. The RightFax fax server supports sending both dynamic fax documents generated by your application and documents previously stored in the fax server's Faxtext library.

The XMediusFAX fax server supports PDF and TIFF document formats. The RightFax fax server supports Microsoft Word®, HTML, plain text, and many other document formats. Because the .NET environment incorporates automation controls for applications such as Word and Excel®, it is possible to create the **fax on the fly** applications whereby a script creates a Word or Excel document and then sends it as a fax. This type of application is ideal for order status applications or other situations where an IVR application faxes dynamic data.

To send faxes through the fax server, you must configure at least one user account on either the XMediusFAX fax server or the RightFax fax server as the sender account.

For complete information on all of the XMediusFAX fax server features, see the *XMediusFAX Integration Guide*. For complete information on all of the RightFax fax server features, see the *RightFax Integration Guide*.

## Creating and Sending Fax Documents

- Create the document that you want to send using a suitable program, and then prompt users for the names of the fax library documents that they want to receive.

**NOTE** Use numeric names for your library documents so that your callers can easily enter the document name from the telephone keypad. If you have a large number of documents, consider placing an index document in your library so that users can fax themselves the index, and use it to select other documents to receive.

For more information about building and organizing a fax library, refer to the *RightFax Integration Guide* or the *Faxtext for RightFax Administration Guide*.

- Create a temporary text file for each fax document to be sent. This text file provides the Fax Server with control information about the fax transaction that the application is currently setting up. The application must give it a unique name to distinguish it from similar files generated simultaneously by other instances of the application.
- Add the following line to the beginning of the text file: **USERID:id & chr(13) & chr(12)** where **id** is the User ID of the user account that you have created for your application on the fax server.
- The script must append the character codes **chr(13) & chr(12)** to the end of this line because the fax server expects the string of characters containing the code to end in a carriage return **[chr(13)]** and form feed **[chr(12)]**.

**For example:**

If you have created a user account with an ID of UC for your UConnect scripts to use when sending faxes, each instance of the script should create a text file starting with the line, **USERID:UC & chr(13) & chr(12)**.

- Add the appropriate control codes to the text file, using a separate line for each one. These codes, which are in the form <control code>, enable you to specify address information such as the recipient's name and telephone number, cover page information such as the sender's name and voice telephone number, the names of all documents to be sent, and other control information. *The RightFax Administrator's Guide* provides a detailed list of all available codes.
- Once you have edited and saved the text file, copy it to the ...\**RightFax\OutGoing** directory on the fax server.
- Create a share for the fax server's disk drive, and then map it on the Call Server so that UConnect scripts can copy files there.
- When the UConnect script copies the text file to the fax server, it renames the file with the file extension, **.tm\$**.
- When you copy the text file to the fax server, rename it with the file extension, **.tmp**. Whenever the fax server detects a file whose name has an extension of **.tmp**, it automatically begins to process it. The additional step of renaming the file while copying it prevents the fax server from starting to process it before it is finished copying.

# Deploying Your Applications

Mitel recommends that you use Visual Studio 2012 or greater to create your scripts on a developer platform, as this ensures that all the needed component imports are available on the Call Server at runtime. If your script uses a non-standard .NET assembly, you must ensure that those non-standard components are on the Call Servers as well.

- You must install and configure the following items:
  - A completed script in the `...\CX\UConnect\Script` directory

**NOTE** For a system licensed for multi-tenanting, the completed script is found in the `...\CX\UConnect\tenant-data\{tenant-id}\Script` folder.

- Any third-party components or .dll files used by your script
- Prompts for your script installed under the `...\CX\UConnect\Speech` folder in a separate folder named for the script name.

**For example:** `...\CX\UConnect\Speech\Lotto`

**NOTE** For a system licensed for multi-tenanting, the script prompts are found in the `...\CX\UConnect\tenant-data\{tenant-id}\Speech` folder.

- You must shut down the UConnect service prior to installing scripts.

**IMPORTANT** Make sure that the installed version of UConnect is the same one under which the script was developed, and that all other scripts running on the Call Server are built using the same version of UConnect.

## Running Scripts on a Remote Server Platform

UConnect scripts can run on one or more remote servers connected to MiCollab AM through a network connection. Mitel recommends that you use multiple remote servers when you need to load-balance the overall system and improve performance. In addition, use multiple remote servers in cases where resource limitations restrict the maximum number of scripts that may run simultaneously on a single platform.

Running scripts on a remote platform is not a recommended practice for most installations since it requires considerably more configuration and setup effort. You may wish to consider alternative methods to improve performance on a sluggish system. For example, use a faster PC, a faster hard drive, or additional memory.

The remote platform communicates with Call Servers using the TCP/IP protocol. The machine name must be the Call Server's domain or host name, or its IP address. Share the drive upon which you installed the MiCollab AM and UConnect software, and then map the shared drive of the Call Server on the remote platform.

The **UConnect Remote Control Panel** utility is included in the UConnect software package. The **Network Name** and **Fully Qualified UNC** boxes default with the information you entered during the setup process, but you can change the information at any time by entering new information, and then clicking **OK**.

The **UConnect Remote Server** utility has the same **Startup** and **Shutdown** functions as the **UConnect** utilities for MiCollab AM and developer platforms, as well as the **Automatic Startup** feature.

On Call Server platforms:

- If you want the application to run only on a remote server, clear the **Launch Scripts on Local Machine** check box in the UConnect utility.
- If you intend to execute scripts from both Call Servers and remote servers, select **Launch Scripts on Local Machine**.

As calls come in, the Call Server distributes the calls automatically based on how you configure the **Launch Scripts on Local Machine** check box. For example, with the box selected, the Call Server gives the first call to UConnect running on the server, and it then sends the next call to the remote server.

## Configuring the UConnect Service for Local and Remote Servers

To allow local and/or remote servers to start UConnect with valid access rights to the Call Server, you must configure the Logon account of the UConnect Server Service on the local, remote, and Call Servers with an administrator's account. This procedure is also required for any other server in the network from which the script needs access to resources.

To set the logon account for the UConnect Server service:

- 1 From the Windows taskbar, select **Start > Control Panel > System and Security > Administrative Tools > Services**. The **Services** window appears.
- 2 Double-click the **MiCollab AM UConnect Server** service. The **MiCollab AM UConnect Server Properties** dialog box appears.
- 3 Select the **Log On** tab.
- 4 Select the **This account** radio button.
- 5 Type an administrator's account name or click **Browse** to search for the administrator's account name.
- 6 In the **Password** and the **Confirm password** fields, enter the correct password, and then click **OK**.

**IMPORTANT** You must delete the dots that are shown in the **Password** and **Confirm password** fields and enter the correct password in these fields before you click **OK**.

- 7 A dialog box advises you that the account is granted the **Log On** service right. Click **OK**.
- 8 Another dialog box advises you to stop and restart the services in order for your changes to take effect. Click **OK** to continue.

- 9 From the **Services** window right-click the **MiCollab AM UCConnect Server** service, and then click **Restart**. The service stops, and then it restarts.

# Setting Diagnostic Logging for UCConnect

The system server provides extensive logging of UCConnect applications through its **Diagnostics** utility, where you can refer to online help topic, *Turning On File Logging*.

**IMPORTANT** Perform this procedure only at the instruction of a Technical Support engineer. Logging may consume large amounts of hard disk storage space and significantly degrade performance.