



Nuance Communications, Inc.
**Nuance Vocalizer for Enterprise
Installation Guide**

Version 7.6.4



Copyright notice

Nuance Vocalizer for Enterprise 7.6.4
Installation Guide

Copyright © 2003-2020 Nuance Communications, Inc. All rights reserved.

Published by Nuance Communications, Inc.
One Wayside Road, Burlington, Massachusetts 01803 U.S.A.
Last updated: Friday, September 25, 2020

Nuance Communications, Inc. provides this document without representation or warranty of any kind. The information in this document is subject to change without notice and does not represent a commitment by Nuance Communications, Inc. The software and/or databases described in this document are furnished under a license agreement and may be used or copied only in accordance with the terms of such license agreement. Without limiting the rights under copyright reserved herein, and except as permitted by such license agreement, no part of this document may be reproduced or transmitted in any form or by any means, including, without limitation, electronic, mechanical, photocopying, recording, or otherwise, or transferred to information storage and retrieval systems, without the prior written permission of Nuance Communications, Inc.

Nuance and the Nuance logo are trademarks or registered trademarks of Nuance Communications, Inc. or its affiliates in the United States and/or other countries. All other trademarks referenced herein are the property of their respective owners.

Contents

Copyright notice	2
Welcome	4
Installing Vocalizer for Enterprise	5
Requirements	5
Hardware requirements	5
Licensing requirements	5
Host requirements	5
Voices	6
Installing Vocalizer	6
Uninstalling previous releases of Nuance Core Services	7
Installing Vocalizer for Enterprise	7
Installing voices	8
Uninstalling Vocalizer	8
Configuring Vocalizer	9
Setting up licenses	10
Configuring the Vocalizer API	11
Installing the Vocalizer client	12
The include, lib, and SO files	12
Configuring an application for the Vocalizer client	12
Configuration parameters	13
Starting Vocalizer	14
Testing the Vocalizer installation	14

Welcome

Vocalizer for Enterprise is a complete engine for delivering spoken output. Available in over 40 different languages with a wide selection of voices, both male and female, Vocalizer can handle all the application's audio, from a library of prompt recordings to dynamically generated text-to-speech (TTS) synthesis. Each Vocalizer voice pack is bundled separately and available for download from Nuance Network.

Vocalizer allows applications to decide the spoken words at runtime (instead of using pre-determined words and sentences), without the intervention of live operators and without the limitations and costs of developing and maintaining prompt libraries.

By providing a single source for all audio output, Vocalizer combines computer-generated and prerecorded audio to enable automation of more application behaviors, deliver detailed information specific to an individual customer, and reduce implementation and operational costs. For example, you can use synthesized speech when developing and testing applications, and then add prerecorded prompts for deployment and tuning. This approach speeds development and reduces the cost of studio recordings.

Other Vocalizer benefits:

- **Avoids unnecessary transfers:** Sometimes customers need to hear dynamic information that is difficult to prerecord. For example, names, addresses, or information from a database. Vocalizer's text-to-speech capabilities can read that information and avoid a transfer to a human agent to complete the task.
- **Automates more calls:** Research shows that callers are more likely to complete an automated call when they hear information such as names and addresses read clearly by a single voice. Callers who hear a mixture of voices and lesser-quality TTS get distracted, and are more likely to leave the automated system.
- **Simplifies application development:** Applications request the desired text and Vocalizer either finds a prerecorded audio file or generates high-quality synthesized speech. When the application requests an assembly of phrases, Vocalizer blends them seamlessly together.
- **Enables complete audio control:** Applications can fine-tune every aspect of generated audio (pronunciations, speaking speed, and so on), which is especially useful for frequently played prompts.
- **Provides superior speech synthesis:** Vocalizer multiform synthesis (MFS) technology and voice models built with recurrent neural network technology produce superior speech synthesis, supported with XPremium-high and XPremium-high-nb voice models.
- **FIPS support:** Vocalizer supports SSL connections to servers that are FIPS-compliant. This protects any data transferred to or from the Vocalizer service (for example, fetched audio files). FIPS refers to Federal Information Processing Standards, a set of USA government standards that includes encryption algorithms.

All components are installed in the Vocalizer installation directory or one of its subdirectories. The `%VOCALIZER_SDK%` environment variable specifies the installation directory (referred to as the *install_path*) for the Vocalizer for Enterprise product.

Installing Vocalizer for Enterprise

You complete these tasks to install and configure Nuance Vocalizer for Enterprise:

1. Verify that the target system meets all hardware and software [requirements](#).
2. [Install the software](#).
3. Create an XML file for configuring licensing and other parameters, as described in [Configuring Vocalizer](#) on page 9.
4. Install and configure a License Manager with license files. You can set up licensing for all licensed products at the same time. See [Setting up licenses](#) on page 10.
5. [Configure the API](#).
6. [Install the client](#). To start Vocalizer, your application must invoke the Vocalizer client by starting a client session using the NVSCClientInit() or NVSCClientInitFromFile() command, as described in [Starting Vocalizer](#) on page 14.
7. [Test the installation](#).

Requirements

This topic describes software requirements for Nuance Vocalizer.

Hardware requirements

The CPU must support the single instruction, multiple data (SIMD) instruction set extension with streaming SIMD Extensions (SSE v4.2).

Licensing requirements

Vocalizer requires a License Manager and license files to deliver them. You must acquire licenses and run license servers. For an overview, see [Setting up licenses](#) on page 10. Alternatively, you can generate licenses separately and merge the resulting license files.

Host requirements

Vocalizer runs on Linux and Windows hosts.

See the *Release Notes* for the supported versions.

Nuance currently tests and supports this product on hosts set to the en-US locale. Running on other locales might affect recognition accuracy and application performance.

User account requirements

You require these operating system accounts to install or remove Nuance speech products:

On Linux:

You must be root or a "sudoer" user. Processes can run under any user member of group "nuance."

Speech Suite services run under the **user nuance** and **group nuance** account. When you use startup scripts provided by Nuance, the processes will start as the **nuance** user (for example, the script *Speech_Server/server/start* or */etc/init.d/nuance-licmgr*).

On Windows:

You must be logged in as Administrator. To run or stop Nuance products, you need an account with proper rights to directory `NUANCE_DATA_DIR` and subdirectories: modify, read, write. This account can be the user Administrator, a user in the group Administrators, or a standard user with these rights for the folder.

Network requirements

Nuance speech products support HTTP 1.1 and HTTPS. If your network uses secure HTTPS, you require these OpenSSL libraries:

- On Windows, Vocalizer uses its own build of the libraries.
- On Linux, Vocalizer uses the libraries supplied with the operating system. The OpenSSL libraries must be installed before installing Vocalizer. If OpenSSL is not present, the Vocalizer installation issues a warning.

Vocalizer dynamically loads the OpenSSL libraries using the `dlopen()` function and the names `libssl.so` and `libcrypto.so`, which are typically symbolic links within `/usr/lib` to the most recent OpenSSL patch release on the system (typically `/lib/libcrypto.so.version` and `/lib/libssl.so.version`).

See the *Release Notes* for the supported versions.

Voices

To run Vocalizer, you must install at least one voice on the Vocalizer host. Install the voice after installing Vocalizer. See [Installing Vocalizer](#) below.

Installing Vocalizer

For an installation overview, see [Installing Vocalizer for Enterprise](#) on the previous page.

Review these notes before starting the installation procedure:

- **Installation host:** You can install Vocalizer on any host; however, you must also install the client on your application host to enable communication with the server. See [Installing the Vocalizer client](#) on page 12 for details.
- **Software location:** The default location is:
 - Linux: `/usr/local/Nuance/Vocalizer_for_Enterprise`
 - Windows: `C:\Program Files (x86)\Nuance\Vocalizer_for_Enterprise`
- **Environment variables:** The installation sets these environment variables. Open a new session to make them available.

Variable	Default	Description
<code>NUANCE_DATA_DIR</code>	Linux: <code>/var/local/Nuance</code> Windows: <code>C:\ProgramData\Nuance\Enterprise</code>	Directory for storing log, cache, utterance files.
<code>VOCALIZER_SDK</code>	Linux: <code>/usr/local/Nuance/Vocalizer_for_Enterprise</code> Windows: <code>C:\Program Files\Nuance\Vocalizer for Enterprise</code>	Installation directory for Vocalizer for Enterprise. (In documentation examples this is what is meant by the Vocalizer install_path .)

Note: If you are running the Vocalizer API, not all required environment variables are set by the installation. See [Configuring the Vocalizer API](#) on page 11 for post-installation configuration.

Uninstalling previous releases of Nuance Core Services

Vocalizer is not compatible on the same host with Nuance Core Services 4.0.

If the host already has an installation of Nuance Core Services, you must uninstall it before installing Vocalizer.

To remove Nuance Core Services:

- On Linux:
 - > `rpm -e NCoreServices-FTA-RES NCoreServices-WATCHER`
- On Windows:
 - Select Add/Remove Programs from the Control Panel.
 - Click on the Nuance Core Services program, and click Remove.

Installing Vocalizer for Enterprise

To install Vocalizer for Enterprise:

On Linux:

1. Remove all previously installed voices and then the software:
 - To remove a voice: `rpm -ve voice`
 - To remove a voice: `rpm -ve voice`
2. Confirm that any other prerequisite software is installed as described under [Requirements](#) on page 5.
3. Download the Vocalizer for Enterprise product from Nuance Network at <https://network.nuance.com>.
4. Extract the software:
 - > `tar -zxvf downloadedFileName.tar.gz`
5. Change to the *Vocalizer_for_Enterprise* directory, and run the *install.sh* shell script.
For example:
 - > `cd Vocalizer_for_Enterprise`
 - > `./install.sh`
6. Follow the online instructions to complete the installation.
7. Open a new session to make `$VOCALIZER_SDK` available to your environment.
8. Install a voice as described in [Installing voices](#) on the facing page.

On Windows:

1. Confirm that any other prerequisite software is installed as described under [Requirements](#) on page 5.
2. Download the Vocalizer for Enterprise product from Nuance Network at <https://network.nuance.com>.
3. Unzip the software.
4. Change to the *Vocalizer_for_Enterprise* directory, and run the *exe* file.
5. Follow the online instructions to complete the installation.

6. Install a voice as described in [Installing voices](#) below.

Installing voices

Note: Do not install voices on network mounted drives.

To install voices, first download the voice from Nuance Network at <https://network.nuance.com> and extract the downloaded file.

On Linux:

1. Change to the *voice* subdirectory under the location where you downloaded the rpm.
2. To install a voice to the default location, *Nuance/usr/local/Vocalizer_for_Enterprise* run the following command:

```
> rpm -i voice.rpm
```

For example, this command installs the American English Samantha voice for Vocalizer for Enterprise under */usr/local/Nuance/Vocalizer_for_Enterprise*:

```
> rpm -i nve-en-US-Samantha-bet1-version#-build#.i686.rpm
```

On Windows:

1. Run the msi program.
2. Follow the online instructions.

The voice is installed by default under *C:\Program Files (x86)\Nuance\Vocalizer_for_Enterprise*.

Installing voices to a different location

On Windows, the Nuance voice installer always installs to the default location. To install to a different location:

1. Unzip the file in the desired location.
2. Set the `VOCALIZER_VOICE_PATH` environment variable to the new location. (To set envvars, click Control Panel→System→Advanced system settings.)

Note that more than one voice model may be available, depending on when the voice was manufactured. (Newer voices have better quality.) If you install more than one kind of voice, say Samantha BET1 and BET4, Vocalizer automatically detects and uses the newest (BET4). To avoid this default behavior and force the use of older voices, use the `voice_model` parameter. (For example, you might want to use an older voice when migrating an existing installation that is tuned for the older voice.)

Uninstalling Vocalizer

To uninstall Vocalizer and voices:

On Windows:

Use the Control Panel to uninstall Vocalizer and the voices. Select a Vocalizer, such as Nuance Vocalizer for Enterprise, and remove it.

On Linux:

To uninstall Vocalizer, use the following command:

```
> rpm -ve nve-api
```


Add these commands if removing all Nuance speech products:

```
> rpm -ve Nuance-Doc
> rpm -ve Nuance-Common
```

To uninstall voices, run this command:

```
> rpm -ve voice
```

For example, to uninstall the American English Samantha voice, use:

```
> rpm -ve nve-en-US-Samantha-bet1-version#-build#.i686
```

Configuring Vocalizer

To set up licensing and other configuration parameters, create and activate a Vocalizer configuration file during installation.

- The file is an XML document that overrides the default configuration settings on the Vocalizer host.
- By convention, the filename *tts_config.xml*, but you can use a different name.

To create the file, follow these general steps:

1. Create a file named *tts_config.xml* in a location outside of Vocalizer's installation path.

The easiest way to create the file is to copy the default configuration in *config\baseline.xml* and delete all parameters except the ones you want to modify.

Sample with description:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<?xml-stylesheet type="text/xsl" href="ttsconfig.xsl"?>
<ttsconfig version="5.5.0"
xmlns="http://www.nuance.com/nvn55/ttsconfig">
  <tts_license_ports>100</tts_license_ports>
  <tts_license_ports_overdraft_thresh>90
  </tts_license_ports_overdraft_thresh>
  <cpr_license_ports>100</cpr_license_ports>
  <cpr_license_ports_overdraft_thresh>90
  </cpr_license_ports_overdraft_thresh>
  <license_servers>
    <server>27000@localhost</server>
  </license_servers>
</ttsconfig>
```

Values in the configuration file can reference variables, specified with the variable name inside "\${...}". For example, for the Vocalizer installation directory, use: \${VOCALIZER_SDK}

The variables can be any environment variable, or any of the special variables discussed in the online documentation.

The file begins with a standard XML declaration:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

Optionally, insert a style sheet declaration for viewing the file in a Web browser:

```
<?xml-stylesheet type="text/xsl" href="ttsconfig.xsl"?>
```

The header includes the root element of the document (specified in the DTD, document type declaration); that is, the container element for parameter elements, <ttsconfig>.

```
<ttsconfig version="5.5.0"
  xmlns="http://www.nuance.com/nvn55/ttsconfig">
```

The main body of the file contains one or more parameter elements, for example:

```
<tts_license_ports>100</tts_license_ports>
<tts_license_ports_overdraft_thresh>90
  </tts_license_ports_overdraft_thresh>
<cpr_license_ports>100</cpr_license_ports>
<cpr_license_ports_overdraft_thresh>90
  </cpr_license_ports_overdraft_thresh>
<license_servers>
  <server>27000@localhost</server>
</license_servers>
```

2. Specify values for the licensing parameter. For a licensing overview, see [Setting up licenses](#) below.
3. Activate the configuration file.
 - o Define the `$VOCALIZER_USERCFG` environment variable and set it to the file's full path including the filename. For example:

```
> setenv VOCALIZER_USERCFG "/home/jruser/tts_config.xml"
```
 - o Specify the location of the file in the command line when starting up the Vocalizer server instance, by setting the `nserver.nvs.VocalizerUserCfgFile` option. For example:

```
> nuance-server -servlet nvs -port 9200
  nserver.nvs.VocalizerUserCfgFile=/home/jruser/tts_config.xml
```
4. (Optional) In the future, insert additional parameters as needed (and restart Vocalizer to load the new values).

Setting up licenses

Vocalizer and other Nuance products require a License Manager with license files. You can set up licensing for all products at the same time using a single server and license file.

Here are the basic requirements:

1. Install License Manager (downloaded and installed separately from Vocalizer).
2. Generate, download, and install a license file from the licensing fulfilment web site. This requires knowing the hostid (Linux) or Ethernet address (Windows) of the License Manager.

To get the ID:

- o **Linux:**

```
> lmutil lmhostid
```

- **Windows:**

- > `lmutil.exe lmhostid`

3. If running the license on a remote host, configure Vocalizer's server list.
4. In a Vocalizer configuration file (*tts_config.xml*), configure Vocalizer's licensing parameters to check out the appropriate number of licenses.

Once set, the licensing parameters only need to change when changing licensing capacity or moving License Manager to a new host.

Parameter	Description
<code>dynamic_license_expiration_time</code>	Sets the expiration timer for relinquishing unused dynamically acquired licenses. The default is 300 (sec).
<code>tts_license_dynamic_ports</code>	Maximum number of full TTS licenses that can be dynamically acquired after exhausting all full TTS licenses acquired by <code>tts_license_ports</code> . The default is 0.
<code>tts_license_ports</code>	Number of full TTS licenses to acquire at startup. Your system needs one license for each concurrent session that uses TTS. The default is 100 (licenses).
<code>tts_license_ports_overdraft_thresh</code>	Threshold at which warnings will be logged to indicate that the system is near the full TTS license limit. The default is 0 (percent).

Configuring the Vocalizer API

Configure these environment variables to ensure the Vocalizer API is correctly installed.

- **Windows:**

Modify `%PATH%` to include the Vocalizer location. For example:

```
PATH=%VOCALIZER_SDK%\common\speech\components
```

- **Linux:**

Modify these environment variables:

<code>\$PATH</code>	Modify <code>\$PATH</code> to include the Vocalizer location: <code>\$VOCALIZER_SDK/common/speech/components</code>
<code>\$LD_LIBRARY_PATH</code>	Modify <code>\$LD_LIBRARY_PATH</code> to include the Vocalizer location: <code>\$VOCALIZER_SDK/common/speech/components</code>

Below is an example using C shell:

To update `$PATH`:

```
> setenv PATH $VOCALIZER_SDK/common/speech/components:$PATH
```

To update `$LD_LIBRARY_PATH`:

```
> setenv LD_LIBRARY_PATH /usr/local/Nuance/Vocalizer_for_Enterprise/common/speech/components:$LD_LIBRARY_PATH
```

Installing the Vocalizer client

To install the client, you must install the full Nuance Vocalizer for Enterprise package.

The include, lib, and SO files

After completing the installation you can find the SO, include, and library files under the *Common* path for your voice platform. These files are required for integrating the Vocalizer client in a custom application.

Linux:

- */usr/local/Nuance/Common/x86/lib/vocalizer-api-client.h*
- All files in */usr/local/Nuance/Vocalizer for Enterprise/api/inc/*
- */usr/local/Nuance/Common/x86/lib/libnvs-client.so*
- */usr/local/Nuance/Common/x86/lib/libvxivalue.so*

Or, if your application is 64-bit:

- */usr/local/Nuance/Common/x86/lib/vocalizer-api-client.h*
- All files in */usr/local/Nuance/Vocalizer for Enterprise/api/inc/*
- */usr/local/Nuance/Common/amd64/lib/libnvs-client.so*
- */usr/local/Nuance/Common/amd64/lib/libvxivalue.so*

Windows:

- All files in *C:\Program Files (x86)\Common Files\Nuance\Common\include*, the *vocalizer-api-client.h* file in particular.
- All files in *C:\Program Files\Nuance\Vocalizer for Enterprise\api\inc*
- *C:\Program Files (x86)\Common Files\Nuance\Common\x86\lib\nvs-client.lib*
- *C:\Program Files (x86)\Common Files\Nuance\Common\x86\lib\vxivalue.lib*

Or, if your application is 64-bit:

- All files in *C:\Program Files\Common Files\Nuance\Common\include*, the *vocalizer-api-client.h* file in particular.
- All files in *C:\Program Files\Nuance\Vocalizer for Enterprise\api\inc*
- *C:\Program Files\Common Files\Nuance\Common\amd64\lib\nvs-client.lib*
- *C:\Program Files(x86)\Common Files\Nuance\Common\amd64\lib\vxivalue.lib*

Configuring an application for the Vocalizer client

To use the Vocalizer client, your application must invoke it via one of two API commands:

- *NVSCClientInit()*—Initiates a Vocalizer client session.
- *NVSCClientInitFromFile()*—Initiates a Vocalizer client session while loading configuration parameters from a prepared configuration file.

These two commands are mutually exclusive: you must use one or the other, but not both. They are used only at application startup, and must be invoked before *TtsSystemInit()*.

To close the Vocalizer client gracefully at the end of the application, your application must use the NVSCClientTerminate() API function. This function must be invoked after the TtsSystemTerminate() function.

Configuration parameters

The Vocalizer client requires certain configuration parameters. These are passed to the Vocalizer client as arguments for the NVSCClientInit() function, or stored in the file whose location is specified in the NVSCClientInitFromFile() function.

- When using NVSCClientInit(), these parameters are expressed as key/value pairs.
- For the NVSCClientInitFromFile() function, you must create a configuration file for the Vocalizer host that lists values for all the required configuration parameters. This file is a simple text file that begins with an [nvs-client] heading, and lists each parameter on a separate line, as shown below:

```
[nvs-client]
server.nvs.Address = localhost:9200
server.nvs-client.Port = 9300
server.nvs.ConnectionTimeout = 5000
```

To disable a parameter, put the # sign at the beginning of the row to comment it out. For example:

```
#server.nvs-client.LogPath = /tmp/nvsclient.log
```

The configuration parameters the Vocalizer client uses are listed below.

Parameter	Default value	Description
server.nvs.Address	127.0.0.1:9200	Specifies the locations of the Vocalizer host with which the Vocalizer plugin communicates.
server.nvs-client.Port	9300	Specifies the port where Vocalizer client connects for implementing the backward B channel.
server.nvs.ConnectionTimeout	5000	Specifies the timeout value (in milliseconds) for connecting (or re-connecting) Vocalizer plugin to Vocalizer host. After this timeout expires, resource allocation will fail and Vocalizer returns negative value into MRCP resource negotiation.
server.nvs-client.LogPath (optional)	[no default]	This is the base log path where diagnostic log file(s) are to be written. If this parameter is not present, no log file is written. When a log file reaches the server.nvs-client.LogMaxSize size, the file is stored in the same folder with a suffix -BCKXX (before file extension, to preserve readability) where XX is a number from 00 to 09. This means that up to 10 backup files may be present on the disk at a given time. The software cycles sequentially through the 10 available log files, overwriting the oldest backup if necessary. For example, if the log file is named <i>log.txt</i> and files <i>log-BCK00.txt</i> to <i>log-BCK09.txt</i> have already

Parameter	Default value	Description
		<p>been created, then when the current <i>log.txt</i> reaches the size limit, it overwrites the backup <i>log-BCK00.txt</i>. When the next <i>log.txt</i> reaches the maximum size, it overwrites <i>log-BCK01.txt</i>, and so on until it cycles through to <i>log-BCK00.txt</i> again.</p> <p>Note: If you have multiple instances of the client, you need to specify different values for this path when specifying it as a parameter of the NVSCientInit() or NVSCientInitFromFile() API function calls.</p>
server.nvs-client.LogMaxSize (optional)	500	This is the maximum size (in MB) allowed for each log file. When this size is reached, the file is rolled over into the next available file.
server.nvs-client.LogLevel (optional)	0	<p>This is the log level desired for diagnostic log file(s). Permitted values are 0, 1, or 2:</p> <ul style="list-style-type: none"> • 0 is for BASIC logs • 1 is for INFO logs • 2 is for DEBUG logs

Starting Vocalizer

To start Vocalizer from the command line, use the nuance-server executable. For example:

```
> nuance-server -servlet nvs -port 9200
```

Your application must invoke the Vocalizer client by starting a client session using the NVSCientInit() or NVSCientInitFromFile() command.

Testing the Vocalizer installation

Use the following utilities to test the Vocalizer installation:

- The nvscmdline utility to verify the installation.

This tool takes as input one text file or input text string and generates an audio file. By default it processes plain text with optional native Vocalizer markup and produces an 8 kHz μ -law audio file. It can also be configured to process SSML and produce a 8 kHz μ -law audio file, or a 8 kHz or 22 kHz linear 16-bit PCM audio file.

For details on how to run this command-line utility, see nvscmdline.

- The nvsload utility to measure Vocalizer performance and scalability.

The Vocalizer SDK contains a set of sample applications, including the nvsload performance test sample application. This tool can be used to measure Vocalizer performance for your specific system configuration. Typically, you specify the channel range for the test, the run duration, and possibly the log level. The runtime can take hours depending on the configuration.