

The Dragon Assistant SDK

Contents

1
1
2
3
4
5
6
6
6
7
8
9
D
0
0
0
0
0
0
1
2

An introduction to the Dragon Assistant SDK

The Dragon Assistant SDK allows developers to add voice functionality to their products that are developed specifically for the target platform.

The Dragon Assistant SDK is a C interface that consists of voice enabled APIs. The Dragon Assistant SDK dll provides access to the Dragon Assistant Core service. Intel will wrap the Dragon Assistant SDK APIs into the Intel Perceptual IDK for use by developers. Developers will use the IDK to access the functions in the Dragon Assistant SDK. In order for an application that uses the Dragon Assistant SDK to run, the Dragon Assistant Core service and at least one language pack must be installed on the target machine.

Overview of Supported APIs and functionality

Developers can use the Dragon Assistant SDK to add the following functionality to their applications:

- Unconstrained voice recognition (dictation) for short utterances (up to 30 seconds)
- Constrained voice recognition (grammar based) for command and control
- Text-to-Speech (TTS) to return an audio buffer (22.05 KHz, 16-bits, mono, raw PCM, little endian). Your application uses the audio buffer to read back the written text to the end-user. Each TTS language is associated with one premium voice.

For more information:

- To learn more about the architecture of the Dragon Assistant SDK and related components, see <u>Dragon Assistant SDK Architectural overview</u>.
- To view more details about the Dragon Assistant SDK APIs and functionality, see <u>Description</u> of APIs and functionality.
- To start using the Dragon Assistant SDK, see Getting started with the Dragon Assistant SDK.

Dragon Assistant SDK Architectural overview

The following diagrams show the relationships between the Dragon Assistant SDK, the Dragon Assistant Core service, and the Intel Perceptual IDK.

Dragon Assistant SDK Integrated Recognition parsing

Audio recognition parsing allows for accurate identification of the user's voice without the need for voice training. Users can say words to launch commands, create dictation, and control Text-to-speech features.



Dragon Assistant application (Desktop)

Applications that use the Dragon Assistant SDK APIs must run on a computer where the Dragon Assistant Core service is installed.

The Dragon Assistant Core service provides the core engine and services for the APIs in the Dragon Assistant SDK.

			Applicatio	n (Desktop)					
				Present Visual an	ation layer nd Dialogue				
UI/Dialog Controller Engine									
<u>Social</u> Facebook Twitter	<u>Media</u> iTunes WMP	<u>VOIP</u> Skype	<u>Search</u> Google Bing Yahoo	Desktop Search Volume Screen Brightness	<u>Video</u> YouTube Hulu	<u>Email</u> Hotmail Gmail/Ymail Outlook	<u>Calendar</u> Hotmail Gmail/Ymail Outlook	<u>Other</u> DMs (disambiguation, confirmation, etc) Persona Configurations Context Sensitive Help	
				Integrat	ion Services				
<u>Social</u> Facebook Twitter	<u>Media</u> iTunes WMP	<u>VOIP</u> Skype	<u>Search</u> Google Bing Yahoo	<u>Desktop</u> Search Volume Screen Brightness	<u>Video</u> YouTube Hulu	<u>Email</u> Hotmail Gmail/Ymail Outlook	<u>Calendar</u> Hotmail Gmail/Ymail Outlook	<u>Other</u> Contact List Retrieval(VOIP, Emails, Social) Calendar Info Retrieval (Calendar) Music/Artist/Title Retrieval (Media) Email, Calendar, Social Notifications	

Intel Perceptual IDK (Desktop)

The Intel Perceptual IDK will provide access to the Dragon Assistant SDK APIs and functionality. The Dragon Assistant SDK APIs will be wrapped into the Intel Perceptual IDK. Developers will use the IDK to create speech-enabled applications.



Description of APIs and functionality

This section contains a high level description of the three APIs, included in the Dragon Assistant SDK, which developers can utilize in their applications. Applications that use the Dragon Assistant SDK must run on a workstation where the Dragon Assistant Core service is installed

For more specific API information, please refer to the API reference documentation provided by Nuance.

Unconstrained voice recognition (dictation) API

Developers can use the unconstrained voice recognition (dictation) API to convert speech to text. The speech must be short (up to 30 seconds) in duration. All strings received from the recognizer are in UTF-8 encoding.

The entire speech utterance must be made using a single language (and not contain words or phrases from multiple languages).

This API is available for languages where embedded dictation exists.

API features

- Ability to query supported languages
- Ability to request recognition in a specific language
- Ability to make the recognizer start listening for audio input
- Ability to make the recognizer stop listening for audio input
- Ability to make the recognizer automatically detect the end of the user's utterance
- Ability to pass an audio buffer (16 KHz, 16-bits, mono, raw PCM, little endian). The API does not use the internal recorder and does not listen to the default recording device.
- After listening and processing audio, set the recognizer to return a String of the recognized text.
- If there is no recognized text or the confidence level of the recognition is low, set the recognizer to return an appropriate error indication.

Constrained (grammar based) voice recognition API

Developers can use the constrained (grammar based) voice recognition API to create a grammar that contains a developer-defined list of elements (words or phrases). In this API, there is no predefined grammar. The developers always have to build a grammar. An end-user says the words in the grammar to perform command and control actions.

All strings received from the recognizer are in UTF-8 encoding.

For speech to be recognized, the end-user must utter the words in the grammar using one language.

API features

- Ability to query supported languages
- Ability to create and build a grammar. You add a list of words and phrases to the grammar.Note: To enable your application to recognize speech, you must create a grammar.
- For dictation, your application can use the internal recorder to listen for audio input from the default recording device. You can also pass and use the audio buffers (16 KHz, 16-bits, mono, raw PCM, little endian).
- Ability to request recognition in a specific language
- Ability to make the recognizer start listening for audio input
- Ability to make the recognizer stop listening for audio input
- Ability to make the recognizer automatically detect the end of the user's utterance
- After listening and processing audio, set the recognizer to return a ranked list of grammar elements and a relative confidence indication for each element ranking.
- If there is no recognized text or the confidence level of the recognition is low, set the recognizer to return an appropriate error indication.

Text to Speech API

Developers can use the Text to Speech API to return an audio buffer (22.05 KHz, 16-bits, mono, raw PCM, little endian). Your application uses the audio buffer to read back the written text to the end-user. The Dragon Assistant SDK is not able to playback the TTS audio.

All strings sent to the TTS engine should be in UTF-8 encoding.

API features

- Ability to query supported languages
- Ability to request TTS in a specific language
- Ability to abort TTS generation in progress. Useful if a request was done for a very long sentence.
- Ability to add in-line formatting to the TTS string in accordance to the Vocalizer Text Markup specification.

Audio input and output

Developers can use the Dragon Assistant SDK to create applications that respond to the enduser's voice without the need for voice training.

The Dragon Assistant SDK features work "out-of-the-box" with the default recording device of the target machine.

The Dragon Assistant SDK integrates with the standard Windows audio interface. This allows developers to influence the audio path and use the audio channel in parallel for alternative purposes.

Applications that utilize the Dragon Assistant SDK can run on systems that are running a Nuance Product - without the applications interfering with each other.

For dictation, your application can use the internal recorder to listen for audio input from the default recording device. You can also pass and use the audio buffers (16 KHz, 16-bits, mono, raw PCM, little endian).

Getting started with the Dragon Assistant SDK

In the following steps, the installation order of the Dragon Assistant Core service and the language pack is not important.

To run an application that uses the Dragon Assistant SDK or to run the provide sample code, perform the following steps:

- 1. Make sure you have met all the requirements outlined in <u>Target Platform configurations and</u> <u>development environments</u>.
- 2. Install the Dragon Assistant Core service.
- 3. Install at least one language pack. For example, Dragon Assistant Language Data en-US 1.0.6.exe.
- 4. Obtain and unzip the Dragon Assistant SDK zip file.
- 5. Install the binary (lib, dll) files.

Review the <u>resources for developers</u> to make sure you have everything you need to start using the Dragon Assistant SDK.

Reach through the API Reference documentation. Look at and run the sample code to learn more about the Dragon Assistant SDK APIs.

Target Platform configurations and development environments

Operating System

- Microsoft Windows 7 Home Basic (32-bit and 64-bit)
- Microsoft Windows 7 Home Premium (32-bit and 64-bit)
- Microsoft Windows 7 Professional (32-bit and 64-bit)
- Microsoft Windows 8 Desktop (32-bit and 64-bit)

CPU

• Intel Core i3 (Sandy Bridge) or better, 2 GHz or faster

Software

- The Intel Perceptual IDK- includes the Dragon Assistant SDK which provides:
- A zip file with:
 - API Reference documentation, including inline code samples
 - Sample code source files
 - C Header files
 - Dragon Assistant SDKdll
- Dragon Assistant Core service installer
- One or more language pack installers: For example, Dragon Assistant Language Data en-US 1.0.6.exe. You must install at least one language pack on the machine that runs your application.

Compiler:

• A current compiler that supports C. The sample code was tested with Microsoft Visual C++2010 Express and Microsoft Visual Studio 2010 Professional.

RAM memory:

• 4 GB or higher

Hard drive size:

• At least 1 GB for each language pack. For example, 6.5 GB for nine language packs

Supported Languages

The Dragon Assistant SDK does not have language dependencies. To enable support for languages, one or more language packs must be installed on the machine that runs your application. For example, if a new language pack for polish becomes available, the language pack for polish must be installed on the machine for your application to support polish.

The language packs provide language support for speech recognition and Text-to-speech. You must install at least one language pack on the machine that runs your application.

Language packs for the following languages are available:

- US English
- Mandarin
- Brazilian Portuguese
- UK English
- European French
- Italian
- German
- European Spanish
- Japanese

Notes:

- Developers can create configurations that only include a subset of the supported languages.
- Speech recognition resources run locally on the target device.

Resources for developers

To enable developers to utilize the Dragon Assistant SDK in their applications, the following items will be available:

- The Intel Perceptual IDK- includes the Dragon Assistant SDK which provides:
- A zip file with:
 - API Reference documentation, including inline code samples
 - Sample code source files
 - C Header files
 - Dragon Assistant SDKdll
- Dragon Assistant Core service installer
- One or more language pack installers: For example, Dragon Assistant Language Data en-US 1.0.6.exe. You must install at least one language pack on the machine that runs your application.