

Speech Recognition Engine

Łukasz Matczyński

Wydział Fizyki, Astronomii i Informatyki Stosowanej UMK

21 grudnia 2011



- 1 Wstęp
- 2 SAPI
- 3 Gramatyka

SPEECH API

- TTS - Text To Speech
- Speech Recognition Engine

Windows 7 posiada **SAPI 5.4**

Configure your Speech Recognition experience



Start Speech Recognition

Start using your voice to control your computer.



Set up microphone

Set up your computer to work properly with Speech Recognition.



Take Speech Tutorial

Learn to use your computer with speech. Learn basic commands and dictation.



Train your computer to better understand you

Read text to your computer to improve your computer's ability to understand your voice. Doing this isn't necessary, but can help improve dictation accuracy.



Open the Speech Reference Card

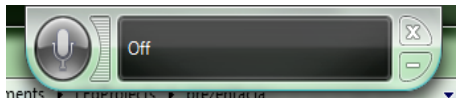
View and print a list of common commands to keep with you so you always know what to say.

Rysunek: Opcje rozpoznawania mowy w panelu sterowania Windows 7

Języki, które potrafi rozpoznawać SRE:

- Chiński
- Japoński
- Angielski
- Francuzki
- Niemiecki
- Hiszpański

potrzebujemy odpowiedniego pakietu językowego.



Rysunek: Windows 7 cię słucha

ISpRecoContext

ISpRecoContext

Główny interfejs rozpoznawania mowy. Tworzony za pomocą metody `ISpRecognizer::CreateRecoContext`

- Każda aplikacja musi mieć przynajmniej jeden `RecoContext`.
- Może mieć więcej.
- Każdy obiekt `ISpRecoContext` może zajmować się innymi zdarzeniami związanymi z rozpoznawaniem mowy.
- Każdy `ISpRecoContext` może używać innej grammatyki.

ISpRecognizer

ISpRecognizer

Interfejs pozwalający aplikacji na wybór odpowiedniego silnika rozpoznawania mowy. Każdy interfejs ISpRecognizer reprezentuje pojedynczy SR engine. Możemy do niego podłączyć wiele kontekstów (ISpRecoContext). Umożliwia wybór źródła audio.

Istnieją 2 implementacje ISpRecognizer'a:

- 1 "shared-recognizer"
- 2 "in-process" (InProc)

SpSharedRecognizer - shared recognizer

SR engine jest tworzony w osobnym procesie. Może być współdzielony przez programy (mogą pracować równocześnie). Jest tylko jeden w systemie. Nie trzeba określać źródła dźwięku, wybiera te ustawione domyślnie w systemie (w panelu sterowania)

SpInProcRecognizer - in process

Tworzony w tym samym procesie co aplikacja. Nie może być współdzielony przez aplikacje. Konieczne jest podanie źródła audio.

Obsługa zdarzeń

W jaki sposób będziemy obsługiwać zdarzenia decydujemy korzystając z `ISpRecoContext` (który dziedziczy z `ISpEventSource`). By określić, które zdarzenia mamy umieszczać w kolejce i o których mamy informować korzystamy z `ISpEventSource::SetInterest` (gdy go nie użyjemy domyślnie będziemy obsługiwać tylko `SPEI_RECOGNITION`).

Najważniejsze zdarzenia

- 1 `SPEI_RECOGNITION`
- 2 `SPEI_FALSE_RECOGNITION`

Gramatyka

Programy do rozpoznawania mowy działają przeważnie na dwa sposoby (tryby):

Gramatyka

Programy do rozpoznawania mowy działają przeważnie na dwa sposoby (tryby):

- 1 *dictation* - program pracuje jako notatnik starając rozpoznać się to co mu powiemy

Gramatyka

Programy do rozpoznawania mowy działają przeważnie na dwa sposoby (tryby):

- 1 *dictation* - program pracuje jako notatnik starając rozpoznać się to co mu powiemy
- 2 *command and control* - program ma określone reguły (słowa), które ma rozpoznać i to ich oczekuje.

dictation

- 1 Tworzymy ISpRecoGrammar:
`ISpRecoContext::CreateGrammar`
- 2 Ładujemy odpowiednią gramatykę:
`ISpRecoGrammar::LoadDictation`
- 3 Ustawiamy stan gramatyki jako aktywny (`SPRS_ACTIVE`):
`ISpRecoGrammar::SetDictationState`

command and control

- 1 `ISpRecoContext::CreateGrammar`
- 2 `ISpRecoGrammar::LoadCmdxxx`
- 3 `ISpRecoGrammar::SetRuleState` or
`ISpRecoGrammar::SetRuleIdState`

Grammar Rule

Zasady określające jakie słowa będziemy rozpoznawać zapisane są w pliku XML. Plik ten zostaje skompilowany za pomocą Grammar Compiler (gc.exe) do pliku *.cfg .

```
1 <grammar LANGID=" 409">
  <DEFINE>
3 <ID NAME=" meeting" VAL=" 1" />
  <ID NAME=" AmA" VAL=" 100" />
5 <ID NAME=" tbre" VAL=" 101" />
  </DEFINE>
7 <RULE NAME=" SCHEDULE" ID=" 139" TOPLEVEL=" ACTIVE">
  <OPT>
9 <L>
  <P>Please</P>
11 <P>Please help me</P>
  </L>
13 </OPT>
  <L PROPNAME=" request">
15 <P VAL=" meeting">schedule a meeting</P>
  <P VAL=" meeting">set up a meeting</P>
17 </L>
  <P>with</P>
19 <L PROPNAME=" participants">
  <P VAL=" AmA">Amy Anderson</P>
21 <P VAL=" tbre">Ted Bremer</P>
  </L>
23 </RULE>
  </GRAMMAR>
```

Wyniki Rozpoznawania

Informacje o rozpoznanym tekście (dla danego `ISpRecoGrammar` i `ISpRecoContext`) znajdują się w obiekcie klasy `RecoResult`, który trafia do nas jako pole struktury `SPEVENT` (znajdują się w niej informacje o zdarzeniu). Rozpoznany tekst możemy odczytać za pomocą metody `ISpPhrase::GetText`.

SPPHRASE

`SPPHRASE` struktura zawierająca informacje o procesie rozpoznawania tekstu w tym hipotezy, błędne rozpoznania, rozpoznany tekst i alternatywnie rozpoznany tekst. Możemy się do niej dostać za pomocą metody `ISpPhrase::GetPhrase`

Kolokwialne Pytania na Kolokwium

- 1 Podaj różnicę między "shared-recognizer" a "in-process".

Dziękuję za uwagę