

PCに於けるGoogle音声認識の利用



Google
日本



D getrunken hatte,
nich." Da segnete
lle! Völker müsse
ht sei, wer dir f
afob hinausgegan

Wind-Craft (風工房)

- 山崎信久 (ウィンドクラフト)
- 船田哲男 (元金沢大)

音声認識の種類

方式	特定話者方式 (特定話者での学習が必要)		不特定話者方式 (学習不要)	
種類	連続認識	単語認識 (限定語認識)	連続認識	単語認識 (限定語認識)
例)	Windows搭載 ドラゴンスピー チ	音声コマンドTool	Julius AmiVoice Google音声認識	Julian(Julius統合)
用途例	口述筆記 翻訳	コマンド操作	口述筆記 翻訳	自動応答

Google音声認識: 単語認識に近いが限定語ではない

音声認識の状況

不特定話者認識ソフト例

・Julius（京都大学等）

○オープンソースで高速な認識

△辞書が固定

学習機能がないため認識率が向上しない



・AmiVoice（アドバンスド・メディア社製品）

○実用的な変換速度と認識率
学習機能があり、認識率向上可能

△入力依存性が高い
ヘッドセット以外では認識率低下
開発者利用は困難



* その他
翻訳ソフトの「超速通訳 ツール」などもある。

AmiVoiceのデモ(メーカ)

The screenshot shows the AmiVoice application window. At the top, there's a menu bar with options like 'ホーム' (Home), '導入' (Introduction), etc. Below it is a toolbar with various icons. The main area displays a Japanese sentence: 'このプロジェクトは、新しい技術を用いて開発されました。'. To the right of the text, there are three buttons labeled 'あア音' (A sound), 'あア音' (A sound), and 'あア音' (A sound). A yellow box highlights the first 'あア音' button. The bottom status bar shows '1/1 文字数: 0' and '日本語(日本) 読みモード'.

- 実用的な認識速度
- 実用的な認識率
- 同音異義語の学習機能
- △入力の依存性大

○実用的な認識率

○同音異義語の学習機能

△入力の依存性大

Googleの音声認識の登場

- インフラの変化
ネットワークの進歩（高速化）
スマートホンの台頭（端末小型化）



- 使用者の変化
通信による情報参照
キーボードレス時の入力手段



- Googleが音声認識を発表
2009年にスマートフォン向けから開始



Google音声認識デモ

○実用的な認識率

キーボードより高速な入力

○入力の依存性小

通常環境で良好な認識

△語句レベルでの認識

長文に不向きだが検索には十分

△認識のタイムラグ

話しながらの認識には不向き

Google音声認識(1)



【経緯】

- ・2009年末Googleが発表
スマートフォン向けとしてiPhone・Android
用に発表

【特徴】

- ・不特定話者認識(学習不要)
- ・実用的な認識率(検索で語彙の予測)

【条件】

- ・ネットワーク接続(要高速)

【ポイント】

サーバで認識する

- ・単語認識に近いが特定語ではない
(良く検索する語句は認識率が高い)
- ・速度は話す速度には追いつかない
(区切りが必要になる)



Google音声認識(2)

【スマートホンの状況】

iPhone: 主に検索用のアプリとして提供されている

Android: 上記以外に

- ・「音声認識_intent(入力ダイアログ風)」として開発利用可能
- ・IME版の開発によりどのアプリでも音声入力可能(ベータ)

google-voice-typing-integration

<https://code.google.com/p/google-voice-typing-integration/source/browse/#git%2FVoicelmeDemo>



【Googleの提供内容】

・iPhone

アプリ内部での利用のみ

(Googleの提供アプリのみの利用になる)

・Android

開発者が利用可能

音声認識を_intent(ダイアログ風)として呼び出し可能

IME版が利用可能

これによってあらゆる入力部分で利用可能

メモでも入力可能・ショートメールにも入力可能

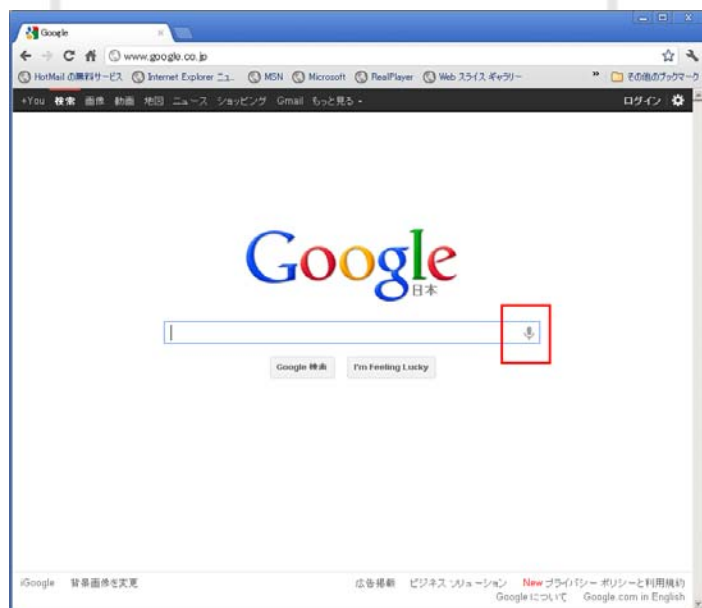


Google音声認識(3)

【PCの状況】

- ・2011年春PC向けChromeに搭載

HTML5の音声入力拡張として動作



音声入力の手法)

HTMLで記述する。(HTML5)

`<input x-webkit-speech type="text" />`
で入力が扱える。

条件)

上のタグを解釈できるブラウザに限る。

現状はGoogleChromeのみ



そのままでは音声検索がしにくいので以下のエクステンションを入れる

Speakable Textareas

<http://userscripts.org/scripts/show/108011>

スマートフォンと同じ環境の実現

GoogleChromeデモ

○スマートフォンと同じ環境

マイクの制限小

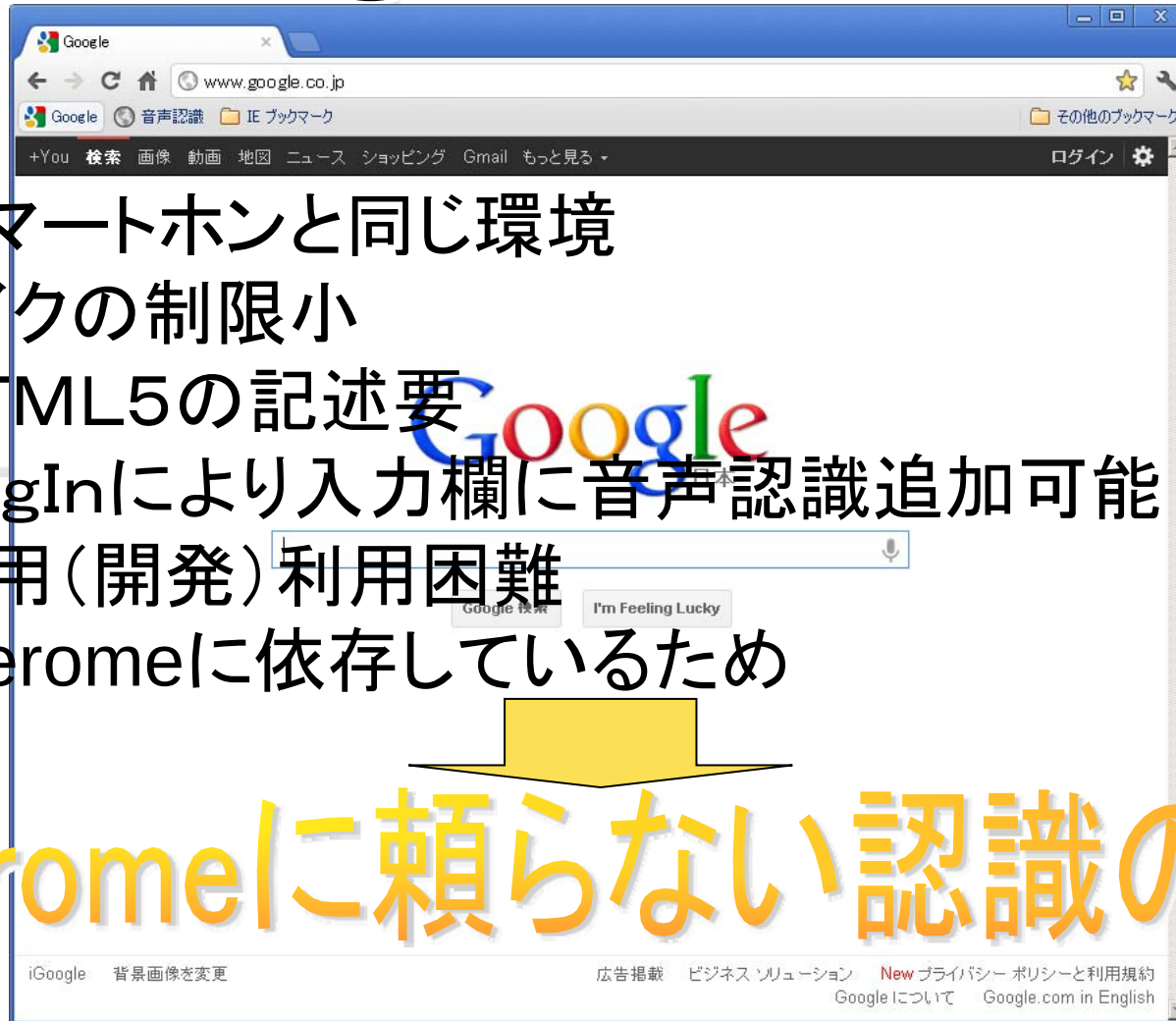
△HTML5の記述要

PlugInにより入力欄に音声認識追加可能

×応用(開発)利用困難

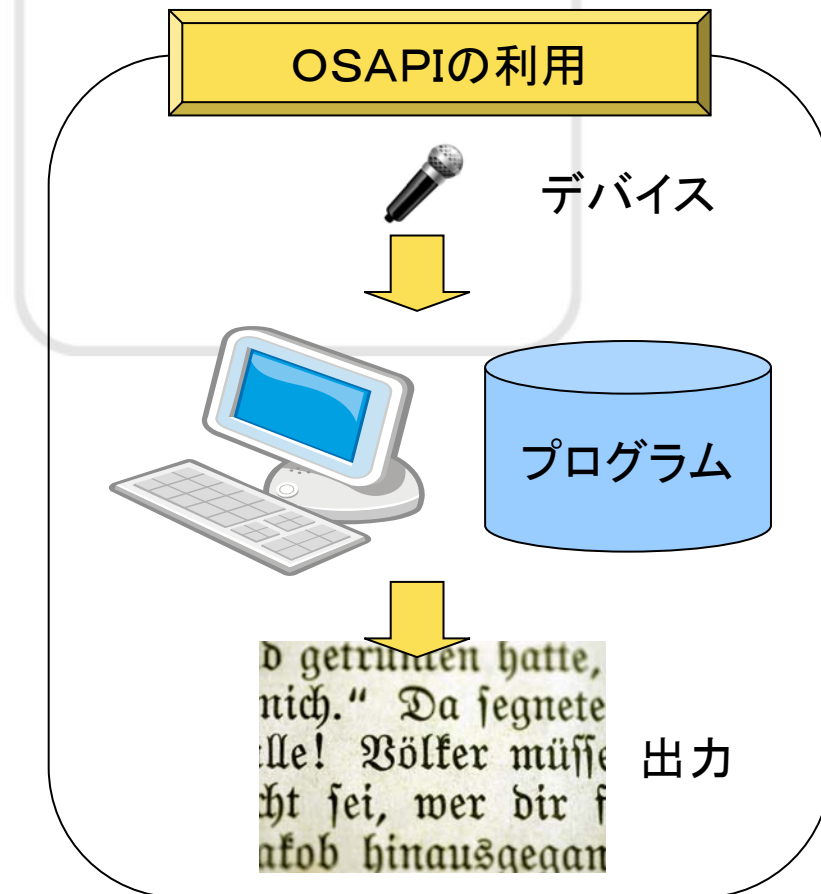
Cheromeに依存しているため

Chromeに頼らない認識の実現



アプリケーションの現状(1)

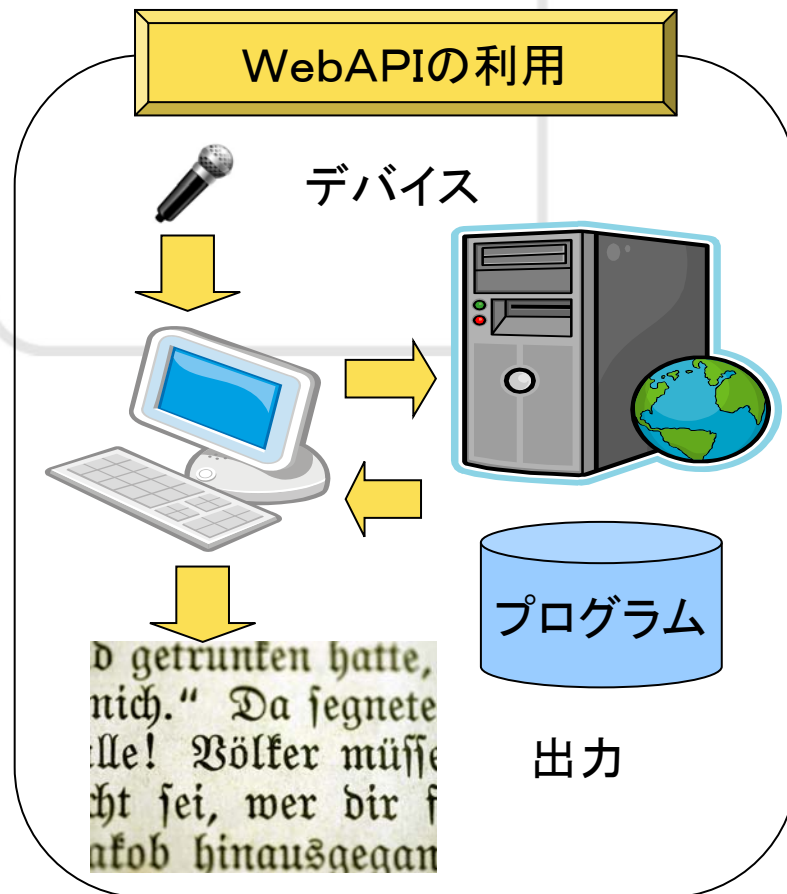
OSAPIからWebAPIへ



- 構成)
処理はすべてOS上のプログラムで処理される。
- 長所)
ネットワーク接続が不要である
- 短所)
・処理能力がPCで決まる
・複雑なプログラムでは多くの知識を必要とする
開発分量が膨大になる
専門知識が必要になる
・OSによる依存性が高い
機種への移植が煩雑

アプリケーションの現状(2)

OSAPIからWebAPIへ



構成)

処理の中核はサーバ上のプログラムで処理される。

長所)

- ・PCの処理能力に依存しない
- ・複雑なプログラムが不要
パラメータを送信し結果を受け取る
- ・OS依存性が低い
HTTP通信があれば可能

短所)

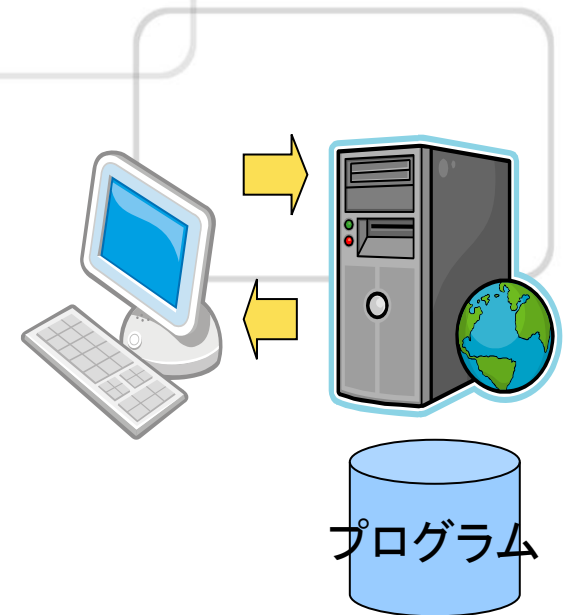
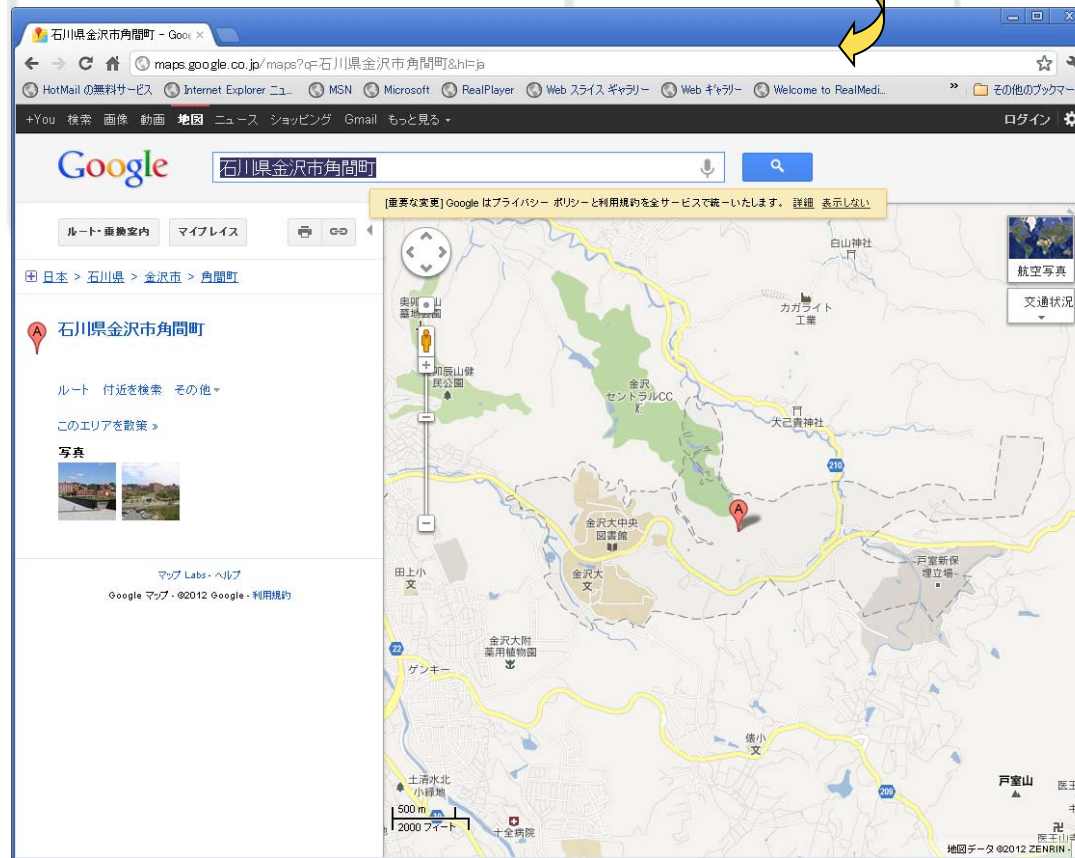
- ・ネットワークが必須になる
パラメータによっては高速必須
- ・中身はブラックボックス
プログラムは相手次第

WebAPIの例(1)

例1)

住所の地図を表示する(金沢大学の住所)

<http://maps.google.co.jp/maps?q=石川県金沢市角間町&hl=ja>

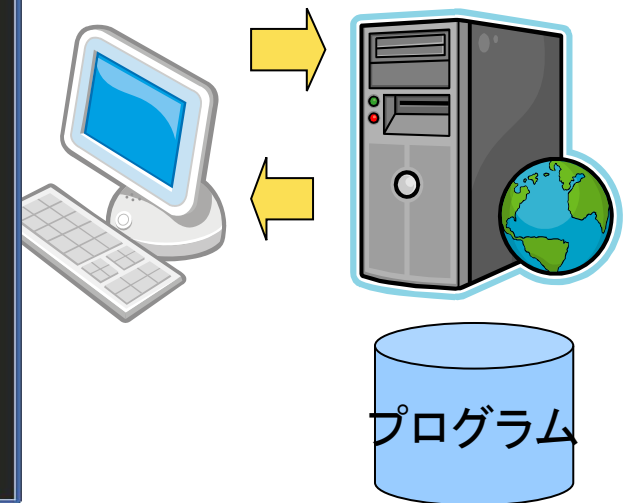
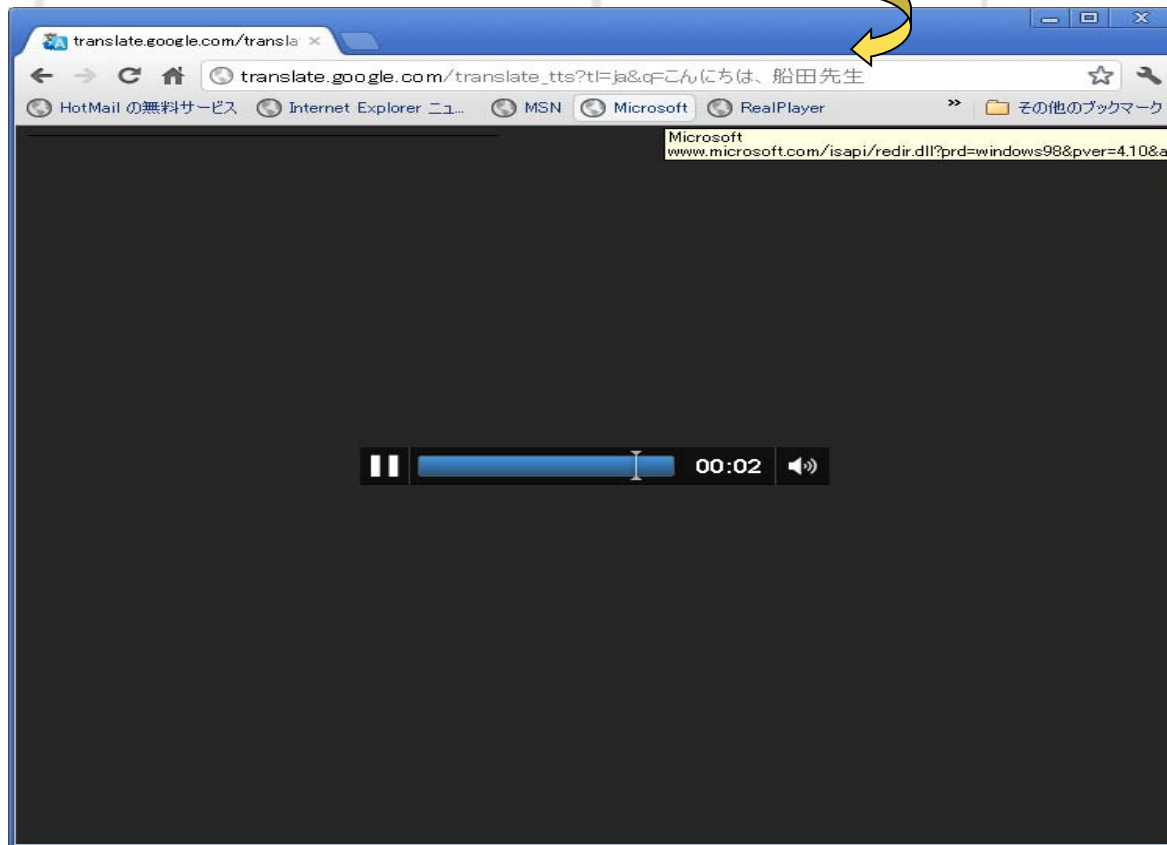


WebAPIの例(2)

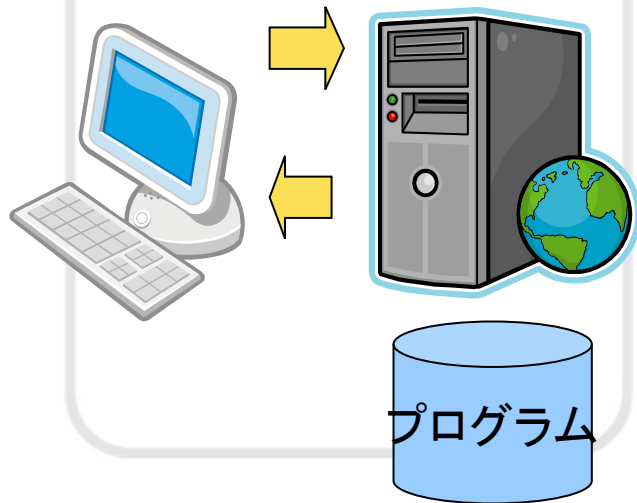
例2)

TTS(テキストの読み上げ)

http://translate.google.com/translate_tts?tl=ja&q=こんにちは、船田先生



WebAPIの使い方まとめ



入力)

GETやPOSTで特定のURLに通知する

出力)

結果として定められた形式で出力される

例)

- ・地図

入力: 住所や緯度経度(座標)など

出力: 画像やマーカ

- ・TTS(読み上げ)

入力: テキスト(+言語)

出力: mp3形式のファイル

【WebAPIの特徴】

- ・基本はブラウザを前提としている

記述はHTMLやJavaScriptで行う。

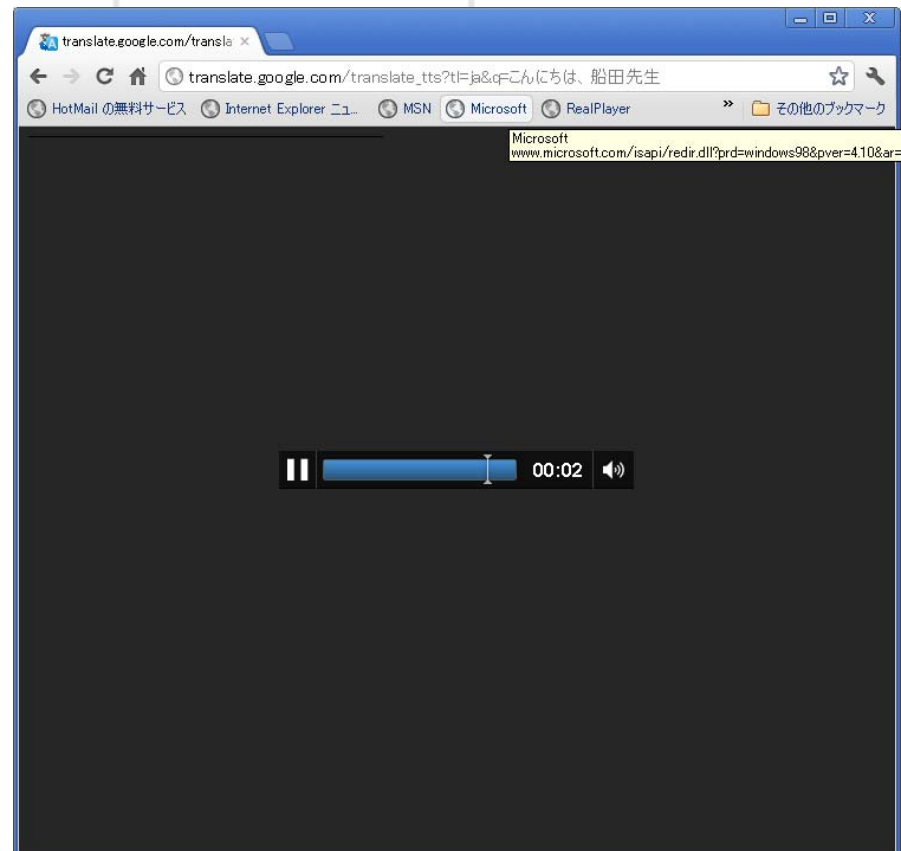
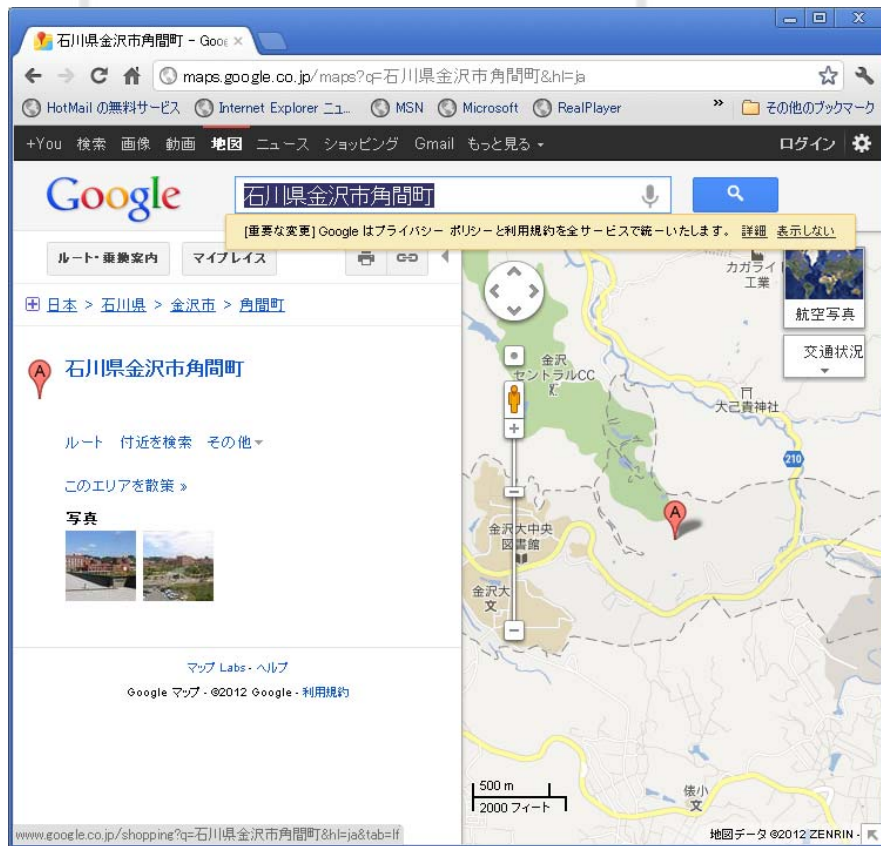
HTMLも時代に応じて拡張がなされている(HTML5)

- ・ローカルPCのプログラムでも

GET/POSTでHTTP通信を行うことでパラメータを送信できる

返答された結果を処理することで、プログラムとして動作できる

WebAPIのデモ



PCに於けるGoogle音声認識(1)

【Chromeの動作分析】

- ・HTML5の構文解釈
- 音声入力→音声をサーバに
入力表示←サーバから結果受信

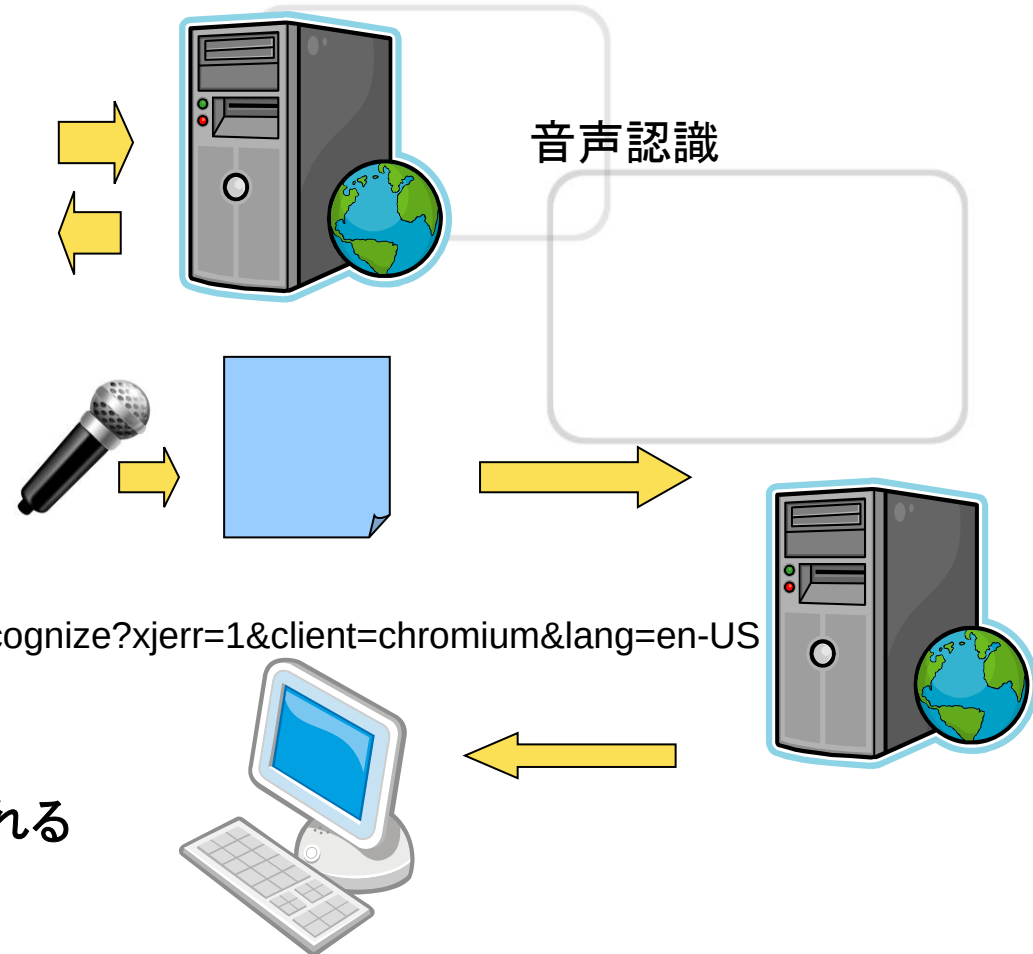
【具体的な内部動作】

- ・マイクから音声データを生成
(音声ファイル化する)

- ・それを以下にPOSTする

<https://www.google.com/speech-api/v1/recognize?xjerr=1&client=chromium&lang=en-US>
(引数に関しては後述する)

- ・認識結果はJSON形式で返答される
(結果の内容に関しては後述する)



原理サイト

<http://mikepultz.com/2011/03/accessing-google-speech-api-chrome-11/>

PCに於けるGoogle音声認識(2)

【技術内容】

1) 音声入力をファイルにする

語句を区切るため「無音検出」で区切る

ファイル形式はflacまたはSpeex形式(flac形式が無難)

サンプリングレートは8KHzまたは16KHz。

16KHzのflacの時Content_Typeは“audio/x-flac; rate=16000”を指定することでGoogleに通知。

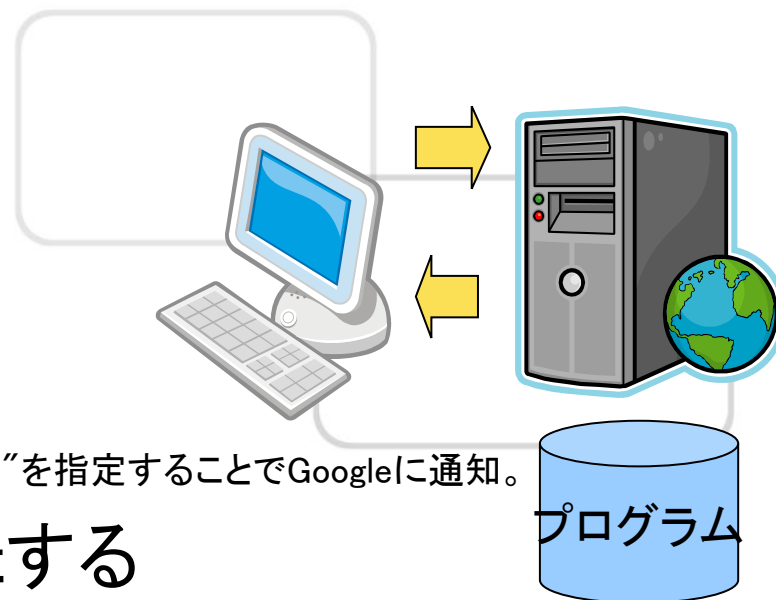
2) 音声ファイルをURLにPostする

引数は言語指定(英語はlang=en-US,日本語はlang=ja-JP)候補数指定(maxresult=10)などがある。

3) 結果をJSON形式で受け取る

結果はstatus(0以外は認識できず)id(順序判定用)に候補指定の分以下が付属する。

hypotheses(予測):utterance(発声)が認識結果、confidence(信頼性)が確からしさ。



解説サイト

<http://sebastian.germes.in/blog/2011/09/>

PCに於けるGoogle音声認識(3)



【応用例】

IEでの音声認識

1) 音声入力ファイル作成

入力からFlashにてwavファイル作成

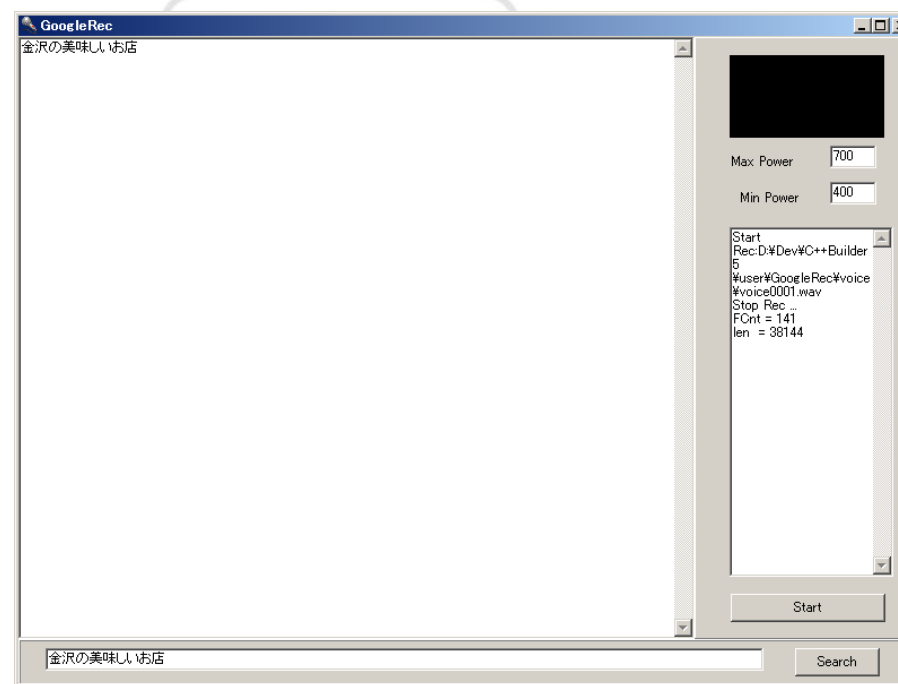
中継サーバにてwav形式をflac形式に変換

2) 音声ファイルのPost

中継サーバからPHPでGoogleにPost。

3) 結果を受け取る

中継サーバで結果のトークンを受け取りFlashに通知。



作成ソフトのデモ

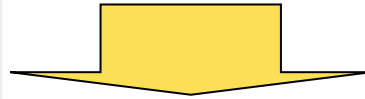
IEデモサイト

<http://select-items.net/pfu/google/speech/>

終わりに(まとめ)

スマートホンの発達(PC+モバイル)

ネットワークの発達(常時接続 & 高速化)



- 機種依存性の少ない開発
HTML+JavaScript
またはそれに類する通信+処理
- WebAPIの活用が加速
通信前提のアプリケーション開発
複雑な内容はサーバがサービス提供
例) 音声認識・音声合成・翻訳・OCR・地図...

