IBM 音声認識製品におけるActiveXを利用したディクテーションサーバの実現

F-031 Implementation of Dictation Server using ActiveX for IBM Speech Recognition Products

友田大輔, 田原義則, 阿竹義徳
Daisuke Tomoda, Yoshinori Tahara, Yoshinori Atake
日本アイ・ビー・エム株式会社 ソフトウェア開発研究所
Software Development Laboratory - Yamato (YSL), IBM Japan, Ltd.

要旨: 音声認識のサーバ製品は、現状ではコマンドをサポートするのみである。筆者らは、ディクテーション（口述筆記）を行うためのサーバを、実現容易性という点で有利と思われるActiveXを利用して試作した。ディクテーションサーバの有用性を技術的側面とビジネスの観点から検証する。

1. はじめに

IBM Viavoice（以下ViaVoice)[1]はデスクトップ環境において音声認識・合成の利用を身近なものにしてきた。現在、その技術はデスクトップからサーバへと利用形態の重点が移りつつある。音声認識・合成を電話回線に適用したIBM WebSphere Voice Server(以下WVS)[1][2]がサーバ製品として発表、発売されている。予約システムや、CTI(Computer Telephony Integration)などのアプリケーションに音声を利用することが主な目的であり、現バージョンではWVSはコマンドのみをサポートしている。今後、高度な会話システムを構築するためにはディクテーション（口述筆記）の利用にまで広がっていくと予想される。筆者らは、今後の音声認識サーバの方向性を検討するためにActiveX[3]を利用したディクテーションのサーバを試作し、その有効性について検証する。

2. 背景

－医療分野での具体的要求

ディクテーションの利用要求のひとつとして、医療分野における電子カルテがあげられる。医療辞書[1]を使用しViaVoiceを利用してディクテーションを行うものであり、したがって現在、デスクトップ上での利用が可能である。クライアント上でViaVoiceを起動し、ダイレクトディクテーションを利用して情報の入力を行う。（図1）

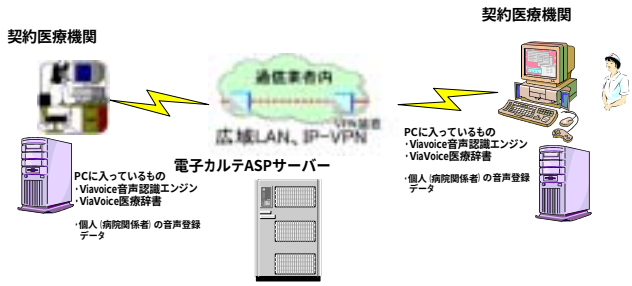

図1. 医療辞書での音声入力

電子カルテはＡＳＰ（Active Server Pages)[4] でのサービスとして利用する形態となっており、音声認識もＡＳＰに統合されることが望ましい。各クライアントマシンにViaVoiceをインストールし、セットアップする手間がいらないというユーザビリティ上の観点からもサーバ側で音声認識を行うことが実際に要求にあがっている。（図2）

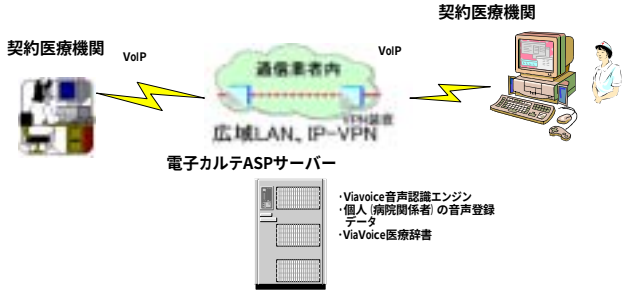

図2. ＡＳＰとしての音声入力

－モバイル環境での利用

一方、音声認識を携帯可能な小さなデバイスでモバイル環境で利用したいという要求もある。Pocket PCなど一部のデバイスではクライアント側での音声認識がすでに可能となっているが、より多くのリソースを使用するディクテーションの実現までには至っていない。十分なリソースを持たないデバイスにおいては、サーバ側で音声認識を行うことによって、モバイル環境でも音声認識を利用することが可能になると考えられる。

3. 試作

今回の作成では、サーバ側にViaVoice Dictation Runtime[1]をインストールすることでディクテーションの認識を行っている。ViaVoiceは通常マイクロフォンからのオーディオを音声認識の入力とするが、独自のDLLファイルを作成することによって認識エンジンへの入力をRerouteすることができる。IPのソケットの入力を認識エンジンへ送るDLLを作成することによって、オーディオデータをStreamとしてクライアントからサーバへ送ればサーバ側で認識を行うことが可能となる。試作ではオーディオデータの圧縮は行っていない。送受信の開始、停止を行うためのマイクボタンをActiveXコントロールとして実現した。ボタンが押された際のアクションとして、オーディオデータの取り込みの開始と停止、ディクテーションサーバへの認識開始、停止のコントロールを行う。

図3に構成と動作の仕組みを示す。クライアントのブラウザでアプリケーションのページを開くと、ページに記述されているActiveXのモジュールがクライアントでロードされ実行される。このActiveXのモジュールは、クライアント上ではマイクボタンのコントロールとして表示されると同時にディクテーションサーバに対して、認識エンジンの接続の要求を行う。次に、ユーザのイベント（クリック）によって、マイクオンの要求を送信し、オーディオデータの取り込みを開始する。ユーザの発声によって取り込まれたオーディオデータは、ディクテーションサーバへIPソケットを通して送信する。ディクテーションサーバから送られるテキストデータをイベントとして受け取り、受け取ったテキストを、クライアントのキーイベントをシミュレートしてターゲットウインドウへと転送する。

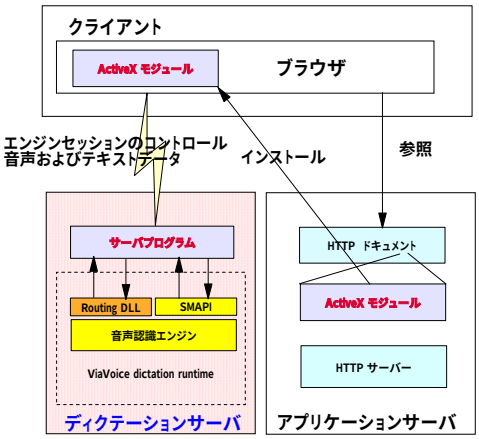

図3. ディクテーションサーバの構成

図4は、試作したサーバにインターネットエクスプローラでアクセスしたところである。マイクボタンを押すとディクテーションが開始され、ページ上のテキスト入力フィールドに認識されたテキストが表示される。

図4. 動作例

以下にサーバの環境を示す。

```
CPU：          Pentium III 1GHz
Memory：       256MB
Network：      100Mbps Ethernet
OS ：          Windows 2000 Professional + SP3
Speech Reco :  ViaVoice V10 Dictation Runtime
```

4. 考察

4．1 Javaとの比較

サーバ上でのサービスを実現するにあたってはプラットフォームの依存を考慮してJavaで実装されることも多い。筆者らは、Javaを利用した場合の検討も行うために、オーディオデータをIP ソケットでサーバクライアント間で転送するプログラムをJavaで作成した。Javaのオーディオサポートを使用するために、JDK1.3以上(またはそれと同等のJRE)をクライアントにインストールする必要がある。また、サポートされているプラットフォームはWindowsおよびSolarisである。ActiveXを使った場合と、Javaの場合とで比較すると、利点と欠点は以下のようにまとめられる。

Java

利点:

- プラットフォーム非依存である

欠点:

- JDK1.3 レベルのものがデフォルトでインストールされている環境が少ない
- 十分使用されていないため、オーディオ処理の品質が十分でない可能性がある。

ActiveX

利点:

- クライアント側のインストールの必要がない（意識する必要がない）
- Windows クライアントと高い親和性があり、細かなコントロールも可能

欠点:

- 対応が Windows のみである

Javaを利用してもサーバでの音声認識の実現は論理的には可能である。実使用環境を考えた場合、Windowsをプラットフォームとして使うケースは多く、クライアントに至っては、Windows環境が大勢を占めるので、プラットフォーム依存になることはそれほど問題ではない。Javaによるオーディオサポートはこれから充実されてくるであろうし、対応するプラットフォームも増えていくであろう。しかしながら、現時点では、可及的ビジネスの対応を行うということでは技術のこなれているActiveXを利用する方が有利に働くものと筆者らは考える。

4．2 応答速度

表1は、ローカル上、およびLANを解した場合でそれぞれ、音声を発声してから最初の単語が認識され表示されるまでの時間を測定したものである。ネットワークを介したディクテーションにおいても実時間で応答が得られ、実用に耐えるものが実現できることがわかる。

	ローカル	LANを経由
応答時間	約1．5秒	約2．0秒

表1．応答時間の測定

4．3 有用性

ActiveXの有用性は、システムをインストールしているという意識がないことにある。ユーザは、目的のページをアクセスするだけで、音声認識が可能になる。ViaVoiceなどの音声認識プログラムをクライアント側にインストールしておき、使用する際に起動するという手間がない。実際に使用してみると、ブラウザを立ち上げ、サーバのページを表示するだけで、マイクボタンが表示される。音声認識がオンデマンドで使用でき、身近に感じられる。

一方、アプリケーションの作成者側からも利点がある。ActiveXとして実現しているので、部品として使用可能である。たとえば、試作したボタンコントロールを音声入力させたいページに貼り付けるだけで、そのページのテキスト入力フィールドは音声入力可能になる。したがって、既存のアプリケーションを音声入力対応にするのも容易である。

ActiveXといえどもサーバ側のプラットフォームはWindowsである必要はない。WVSでディクテーションがサポートされれば、サーバとしてWVSを使用することが可能となり、電話回線を通じてのディクテーションが可能になる。逆に、WVSにディクテーション機能を追加することによりビジネスのチャンスが広がることも意味しているといえるであろう。また、ActiveXはWindows CE でも動作するので、クライアント側としてWindows CEを採用したPDAや、車載のシステムなどでも実現可能である。

5. おわりに

今回のActiveXによるディクテーションサーバの試作は、オーディオデータの受け渡しによるテキスト表示の部分だけであり、本質的な部分ながら非常に簡単なものである。デスクトップ、電話はもとより、モバイル環境で使用するような小さなデバイスでも、高い音質の得られるオーディオ装置と、IPの接続できる環境であれば、実アプリケーションの利用のためにサーバを実現することが可能であることが確かめられた。今後の課題として、オーディオのセットアップ、エンロールをどう実現していくか、また、翻訳サーバなどの付加価値をつける他のサーバとの連携の実現可能性を探っていくことがあげられる。

謝辞

医療関連サポートにおけるお客様の要求に関する情報提供をしてくださった日本アイ･ビー･エム（株）ソフトウェア事業部、WebSphere技術の塩谷充さんに感謝します。

参考文献

[1] ボイスらんど, http://www-6.ibm.com/jp/voiceland/

[2] Websphere Voice Server V3.1 の発表, http://www.ibm.com/news/jp/2002/09/09261.html

[3] Tom Arnstring著, 柏原 正三訳, "ActiveXコントロールプログラミング", アスキー出版局,1997

[4] ASP入門, http://asp.dataweb.ne.jp/