

# WIPI기반 음성 웹브라우저 구현 방안

유세영, 김병기

전남대학교 소프트웨어공학협동과

## Strategy for Implementing A Voice Web Browser Based WIPI

Seyoung Yu, Byungki Kim

Interdisciplinary Program of Software, Chonnam National University

E-mail : [panavia@hanmail.net](mailto:panavia@hanmail.net)

### 요 약

인터넷 및 휴대폰들이 일반화되고 음성처리 기술이 실용화 단계로 발전함에 따라 음성 응용분야가 새로운 이슈로 떠오르고 있다. 음성처리 기술은 사람의 말을 알아들을 수 있는 귀와 사람에게 말을 할 수 있는 입을 마련해주는 새로운 분야다. 그리고, 음성으로 웹의 콘텐츠를 개발하기 위한 표준 언어인 VoiceXML, SALT가 빠르게 보급되고 있다. 음성인식과 음성합성 기술이 꾸준히 발전하여 음성 포털 서비스나 자동 음성 안내 시스템 등에 음성인식과 음성합성 기술이 채택되는 등 상용화 수준에 이르렀다. 사람에게 가장 편리한 정보 습득 방법은 음성이고 이러한 음성을 적용한 음성 웹 브라우저를 현재 유선 상에서 사용하고 있다. 하지만 아직까지 무선 플랫폼에 적용하여 사용하는 브라우저는 개발되지 않고 있다. 사용자에게 친숙한 무선인터넷 환경을 제공하고자 무선 음성 웹 브라우저를 구현방안을 제시하고자 한다.

## 1. 서 론

현재의 무선 인터넷 기술 발달의 속도는 유선 인터넷 발전과정보보다 더 빠르게 진화하고 있고 그중 휴대폰이라는 모바일 단말기는 음성 및 문자서비스 위주에서 고기능화 기술의 진보로 동영상 및 사진 대용량 데이터 전송에 까지 영역을 확대하는 종합적인 커뮤니케이션 미디어로 발전을 계속하고 있다. 또한, 앞으로는 인간과 인터넷을 연결시켜주는 새로운 도구로서 급속도로 진화해 나갈 것으로 보인다.

따라서, 무선 인터넷의 성공여부는 모바일 기기의 특성을 최대한 강조 할 수 있으면서 불편함을 최소화 할 수 있는 방안을 사용자에게 찾아주는 것이 첫 번째 열쇠라고 할 수 있다.

인터넷 휴대폰들이 일반화되고 음성처리 기술이 실용화 단계로 발전함에 따라 음성 응용분야가 새로운 이슈로 떠오르고 있다. 음성으로 액세스 할 수 있도록 웹의 콘텐츠를 개발하기 위한 표준 언어인 VoiceXML, SALT가 빠르게 보급되고 있기 때문이다.

사람에게 가장 편리한 정보 습득 방법은 음성이고 사용자에게 보다 친숙한 무선인터넷 환경을 제공하고자 무선 음성 웹 브라우저를 구현방안을 제시하고자 한다.

## 2. 음성 처리 기술

음성인식이란 음성이 담고 있는 정보를 추출해 컴퓨터를 통해 음성의 의미를 알아내는 기술이다. 이는 크게 세 가지 측면에서 분류되는 데 먼저 적용화자에 의한 분류로는 화자 종속 음성인식 기술과 화자 독립 음성인식 기술이 있으며, 발음 방식에 의한 분류로는 고립단어 음성인식 기술, 연결단어 음성인식 기술, 핵심어 인식 기술, 연속 음성인식 기술 그리고 대화 음성인식 등 5가지 기술로 분류된다. 마지막으로 인식 어휘에 따른 분류로는 소용량 어휘 인식 기술, 대용량 어휘 인식 기술, 가변 어휘 인식 기술 그리고 어휘 독립 인식 기술이 있다.

음성합성(TTS: Text-To-Speech)이란 문자 정보 또는 기호를 인간의 음성으로 변환하여 들려주는 기술이다. 이것은 한국어의 모든 음소에 대한 발음 데이터베이스를 구축하고 이를 연결시켜서 연속된 음성을 생성하게 되는데, 이때 음성의 크기, 길이, 높낮이 등을 조절해 자연스러운 음성을 합성 해낸다. 음성합성 기술은 응용 방식에 따라 제한 어휘 합성과 문자-음성 합성으로 구분 될 수 있다.[1]

하지만, 음성인식 측면에 문제점으로 현재의 음성인식 기술에서 동시에 여러 사람이 말을 하는 경우, 누구에게 말을 하는지 아니면 어떠한 말

을 인식해야 하는지 컴퓨터는 알 수 없으며, 현재 많은 연구가 진행되고 있는 분야이지만 아직까지는 자연언어의 처리에는 한계가 있다.

음성합성 측면 문제점은 음성합성 기술이 많은 발전을 하여 거의 자연스러운 음성합성이 되지만, 전체 문장의 의미와 억양, 속도 등이 규칙적이기 때문에 일반 처음 사용자들은 잘 알아듣기 힘들다. 음성 합성기가 특수문자를 인지하지 못하여, 불필요한 발음을 연속적으로 할 수가 있으며 방대한 음성 DB의 사용과 많은 연산을 사용하여 높은 사양의 시스템을 요구한다.

위의 문제점들이 현재 음성 웹 브라우저를 실행하는데 있어 음성인식에 문제점으로 분류할 수 있다.

### 3. 무선 음성 인터페이스 구현 방식

음성 인터페이스를 구현하는 방식으로 크게 세 종류가 존재한다. 첫 번째 방식은 단말기에 음성 인터페이스를 구현하는 방식이다. 즉 음성인식, 음성합성 및 단화 기술이 단말기에 구현되고 사용자는 단말기와 자연스럽게 음성으로 명령을 내리거나 명령 결과를 소리로 듣게 되는 방식이다. 이 방식은 단말기의 성능 및 메모리를 많이 요구하며 특정 단말기에만 가능하므로 제조 회사 중심으로 채택되고 있다.

두 번째 방식으로는 서버에 음성 인터페이스 기능을 제공하고 단말기는 단순히 연결만 해주는 방식이다. 이 방식은 전화망을 이용하는 음성 인터페이스에 많이 사용하고 있는 방식이며 주로 통신 서비스 회사에서 추구하는 방식이다. 그러나 음성 인터페이스를 사용하기 위해서는 음성 채널을 사용해야 하므로 다양한 단말기에 활용하기 어렵다.

세 번째 방식으로 하이브리드 방식이다. 즉 서버와 단말기기의 성능 상태에 따라 음성 인터페이스를 단말기 혹은 서버에 나누어서 구현하는 방식이다. 이 방식은 음성 인터페이스를 위한 특정 추출 등과 같은 단순한 일은 단말기에서 하고 복잡하고 시간이 걸리는 일은 서버에서 하도록 하는 방식이며 최근에 표준화로 연구되고 있는 분야이다[2].

유선환경의 음성 인터페이스는 단말기에 직접 음성 인터페이스를 구현하는 방식이 유용하지만 무선 환경에서의 음성 인터페이스는 단말기의 성능이 아직까지는 일반적인 PC성능에 미치지 못하므로 두 번째 방식인 서버에서 음성 인터페이스 기능을 제공하고 단말기는 단순 연결만 하는 방식을 사용하는 것이 좋다. 하이브리드 방식도 무선 환경에 적합할 수 있으나 단말기의 성능에 따라 음성 인터페이스 구현이 달라지므로 본 논문에서는 두 번째 방법을 사용하여 구현하는 방안을 제시하고자 한다.

### 4. VoiceXML & SALT

VoiceXML은 웹 기반의 음성 인터페이스를 지원하고 개발 할 수 있는 대화형 마크업 언어로, 음성 인식과 음성 합성과 같은 전문적 기술에 대한 지식이 없어도, 사용자는 대화형 음성 정보 서비스에 대한 시나리오를 구성하는 데에만 집중하면, 대화형 음성 정보 서비스를 쉽게 제공할 수 있도록 구현하는데 매우 유용하고 중요한 기술이다.

또한, VoiceXML 은 대화형 음성서비스의 인터넷 기반 개발을 위한 문서형 인터페이스로 XML spec 1.0에 기초하여 html 이 웹 페이지를 만드는 데 사용되는 것처럼 VoiceXML 은 음성 서비스를 정의하는데 사용되는 마크업 언어 이다. 웹 기반의 대화형 마크업 표준으로 음성을 통한 사용자 인터페이스를 지원하고 개발할 수 있도록 만들어졌다. 즉 VoiceXML은 음성출력, 음성명령 및 DTMF 신호를 인터넷 기반 웹 상에서 제어 하도록 해 주는 XML(Extended Mark-up Language) 언어이다. 특히 전화망을 통한 음성 입·출력을 웹을 통한 표준화 언어로 제어 해 주는 방식이다.

SALT 는 음성의 상호작용을 마크업 하는 언어이다. SALT는 관련된 속성과, DOM 객체의 속성과 함께 사건과 방법을 가진, XML 의 요소의 작은 집합으로 구성되고, 여러 웹 페이지와 음성의 상호작용에 적용할 수 있으며, HTML, XHTML, 다른 표준의 양쪽의 음성만을 중계하는 음성쓰기 그리고 복합 응용과 함께 사용할 수 있다. SALT 는 결과로써 어떠한 개개의 마크업 언어도 직접적으로 확장하지 않는다. 다른 마크업 언어를 통해 확장되어 분리된 계층과 같은 음성 인터페이스를 적용한다.

SALT는 개발자들이 음성 및 멀티모달(Multimodal) 응용제품을 쉽게 개발 할 수 있도록 HTML, XHTML, XML 같은 웹 표준 언어를 확장한 것이다. 2000년 10월에 MS를 주축으로 시스코, 인텔, 컨버스, 필립스, 스캔소프트, 스피치웍스 등의 회사들이 모여서 전화, PC, 노트북 PC, PDA를 이용해서 음성, 펜, 마우스를 사용한 입·출력 명령을 제어할 수 있도록 한 멀티모달 인터페이스이다. VoiceXML 기술이 전화망 기반 음성 응용 개발자들을 위한 상위 수준의 언어라면 SALT는 웹 기반 개발자들을 위한 하위 수준의 언어이다[3][4].

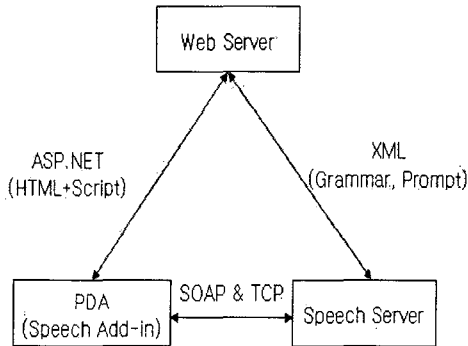
본 논문에서는 웹 기반 음성 인터페이스 구현 방식을 이용한 무선 환경의 웹 브라우저 설계를 하기 위한 방법을 제시하기 때문에 VoiceXML보다는 SALT기반 음성 인터페이스를 사용하고자 한다.

[표 1] VoiceXML과 SALT 비교

언어 비교	VoiceXML	SALT
목적	· 대화형 음성 언어 인터페이스 제공	· 플랫폼에 독립적인 개방형 표식언어 제공
장점	· 다양한 서비스 및 콘텐츠 제공 · 기존 웹 기반 기술 활용 · 클라이언트와 서버 상호작용 최소화	· XHTML과 같은 표식언어의 하위 그룹으로 웹컨텐츠에 음성인식, 합성 콜 컨트롤 기능 추가 가능 · 핵심언어의 사양을 특정 기기의 기능의 프로파일과 결합

### 5. Speech Server를 이용한 무선 음성 웹브라우저 설계

[그림1]은 Speech Server에는 무선단말기인 PDA에 필요한 음성엔진이 적재되어 있고 추가적으로 필요한 엔진이 적재 될 수 있음을 나타낸다.



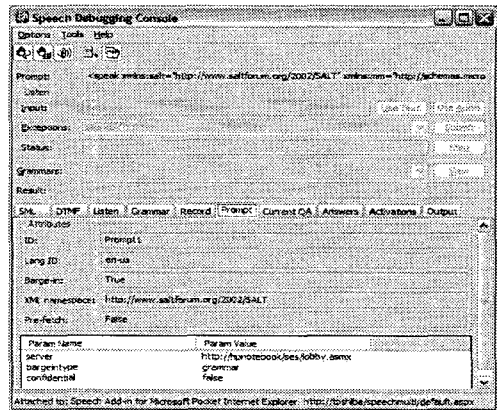
[그림 1] 무선 서버 구성도

웹 서버는 SALT로 웹 페이지를 변환시키고 Speech Add-in for Pocket Internet Explorer는 어느 것이 Pocket PC에 설치되어야 하는지 SOAP를 사용함으로써 Speech Server와 선택을 시작한다. 웹 페이지는 사용자로부터 음성 입력을 요구하거나 TTS 출력을 하게 된다. 음성을 입력할 때 사용자는 Pocket PC 마이크로폰을 통해 음성을 기록하기 위해 버튼을 누르게 되고 Pocket Internet Explorer Speech Add-in은 음성 입력을 처리하고 기록된 음성을 압축한 Speech Server로 TCP 전송을 한다. Speech Server는 전송된 음성은 문법에 따라 처리하고 XML로 결과를 보낸다.

Speech Add-in은 웹 페이지에 결과를

Display 해주고 웹 페이지가 TTS 출력을 요구하면 Speech Add-in은 Speech Server에 텍스트를 보내서 Speech Server에서 음성 변환 후 압축하여 Pocket PC에 전송한다. Pocket PC는 전송된 내용을 스피커로 보내기 위해 TCP로 합성된 음성을 보내 복원시켜 준다. 이와 같은 방식은 단지 이동 단말기에만 적용 된다. PC는 음성 인터페이스 구현 시에 하드웨어적으로 음성 인식/합성 엔진을 내장 할 수 있지만 이동 단말기는 그것이 불가능하기 때문에 위의 방식을 따르는 것이고 만일 PC에서 위와 같은 방식을 구현하고자 한다면 Speech Server는 불필요 할 것이지만 Speech Add-in은 꼭 필요하다. 그 이유는 웹 페이지에서 읽어 들인 SALT를 해석하고 음성 엔진에 필요한 파일 변환을 해주기 때문이다.

[그림2]는 PDA와 Speech Server가 연결되어 Speech SDK의 Speech Debugging Console을 띄우면 위와 같이 연결된 모습을 볼 수 있다. 또한, PDA와 Speech Server 그리고 Web Server의 동작 상황과 현재 진행 정도까지 볼 수 있다.



[그림 2] 실행 화면

웹 페이지 작성을 위해서 마이크로소프트사의 비주얼 스튜디오 닷넷을 사용할 수 있는데, 여기에 Speech SDK를 설치하면 손쉽게 음성 웹 페이지 제작이 가능하다. 그러나 이를 위해서는 SALT와 ASP.NET, JavaScript, GrammarXML, C#언어를 알고 있어야하고 이를 복합적으로 사용해서 프로그래밍 했을 때 음성 웹페이지가 작성된다.

SALT가 VoiceXML로 웹 페이지를 작성할 때보다 복잡한 이유는 SALT는 VoiceXML보다는 하위 언어에 속하지만 VoiceXML보다는 섬세한 프로그래밍이 가능하다. 또한 VoiceXML처럼 웹 페이지와 음성페이지를 따로 제작하는 것이 아니라 ASP.NET페이지에 모든 것을 포함시킬 수 있다. 단순히 유선상의 웹 페이지를 작성한다면 PC에서 손쉽게 테스트가 가능하고 VoiceXML페이지처럼 VoiceXML게이트웨이를 만들어줄 필요가 없다 [5][6].

SALT 프로그래밍은 VoiceXML 프로그래밍과 유사점이 많다. 단지 SALT프로그래밍에서 GrammarXML과 JavaScript는 ASP.NET페이지 안에서 작성되고 서로 독립적으로 움직이는 것이 아니라 종속적이라는 점이 다르다. 또한 JavaScript가 할 수 없는 프로그래밍을 C#이 보완해서 좀 더 많은 작업을 할 수 있다.

```

<html>
.....
  <speech:QA id="Rule1" runat="server">
    <Prompt InlinePrompt="Select the information
    toward among the news, weather, traffic."
    BargeIn="false">
      <Params>
        <speech:Param Name="bargetype">
          ... .. </speech:Param>
        </Params>
      </Prompt>
      <Answers>.....
        <speech:Answer SemanticItem="siR1"
        XpathTrigger="/Rule1"></speech:Answer>
      </Answers>
      <Reco InitialTimeout="5000"
      BabbleTimeout="3000"
      EndSilence="1000" MaxTimeout="30000">
        <Grammars>
          <speech:Grammar
            Src="Grammar1.grxml#G1">
            </speech:Grammar>
          </Grammars>
        </Reco>
      </speech:QA>
    </html>
  
```

[그림 3] SALT프로그래밍 관련 소스 예

## 참고 문헌

- [1] 임채욱 장민석, "모바일 환경에서 VoiceXML 기반의 VUI 개발에 관한 연구", 『한국정보처리학회 논문지』, 제10권 제1호, pp.539-542, 2003년 5월.
- [2] ITFND 주간기술동향, "음성 인터페이스와 멀티모달인터페이스", <http://kidbs.itfind.or.kr/new-bin/WZIN/WebzineRead.cgi?recno=0901014037&mcode=jugidong>
- [3] Speech Application Language Tags(SALT) Forum, <http://www.saltforum.org>
- [4] Kuansan Wang, "SALT: A SPOKEN LANGUAGE INTERFACE FOR WEB-BASED MULTIMODAL DIALOG SYSTEMS." ICSLP, 2000.
- [5] Multimodal Speech on a Pocket PC, <http://www.brains-n-brawn.com>
- [6] 마이크로소프트 스피치 <http://www.microsoft.com/speech>

## 6. 결론 및 향후 연구

무선 환경은 유선과 프로그래밍이 유사하지만 몇 가지 고려해야할 점이 있다. 그중 하나가 SOAP을 이용한 통신이다. 유선은 일반적으로 PC에서 수행되기 때문에 하드웨어에 음성 엔진이 적재 되지만 무선은 그렇지 못하기 때문에 Speech Server에서 음성과 관련된 작업을 해주어야 하고 결과를 SOAP를 이용해 이동 단말기에 전송해 주어야 한다. SOAP는 Speech Add-in에서 관련된 작업을 수행해 준다고 하지만 Speech Server에서 이동 단말기와 통신을 하지 못하는 음성을 이용한 통신이 불가능 문제점이 발생한다. 앞으로의 연구 수행은 이러한 Speech Server와 이동 단말기간의 통신수행을 위한 문제점 개선과 Speech Add-in을 이용한 통신을 보다 효율적으로 수행할 수 있는 방법과 웹 서비스를 통한 음성 서비스를 연구 하여야 할 것이다.