

HanPhone Server: A Technical Overview

1. Introduction

HanPhone Server is a powerful and flexible voice application platform developed by KanHan Technologies. It encapsulates computer telephony hardware and TTS (text-to-speech) software, allowing developers to build voice applications using a subset of HTML as well as a simple markup language called HanPhoneXML. Using HanPhone Server, the process of developing an IVRS (interactive voice response system) is very similar to that of developing a web application.

In addition to making the development of voice applications easier, HanPhone Server also greatly simplifies the deployment of these applications. HanPhone Server only requires Java Runtime Environment version 1.4 (or above) on a machine running Microsoft Windows 2000 Server (or above) and equipped with Dialogic (or compatible) voice boards. Compared with proprietary IVRS hardware and software, the cost of deploying and maintaining HanPhone Server is low.

2. The role of HanPhone in a voice application

As mentioned earlier, the process of developing a HanPhone voice application is very similar to that of developing a web application. In both cases, the application logic resides on the content server and determines what is delivered to the user. This is typically implemented with server-side programming languages such as PHP, ASP and JSP in conjunction with a database storing user information and content.

In a web application, the browser on the user's computer renders the content in the form of HTML from the content server:

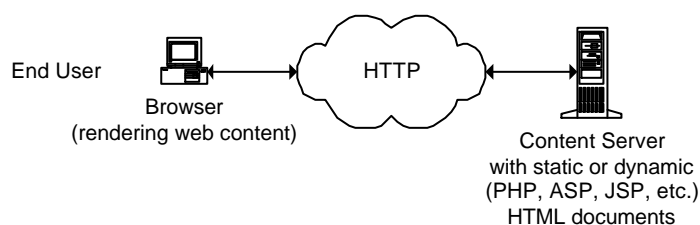


Figure 1. Web application

On the other hand, in a HanPhone voice application, the HanPhone server renders the content in the form of HanPhoneXML or HTML from the content server:

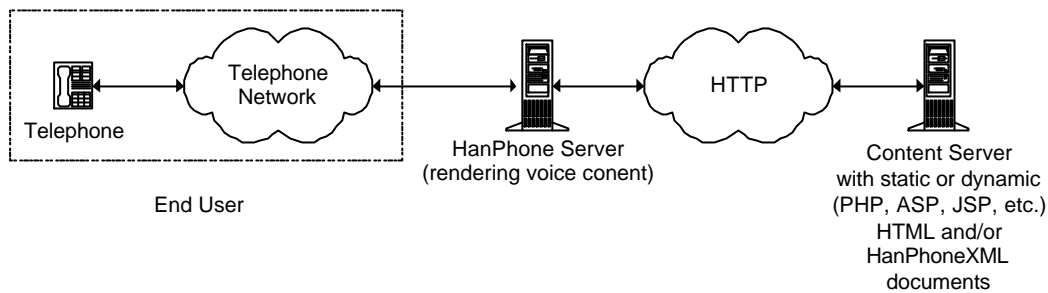


Figure 2. HanPhone voice application

Figure 1 and 2 show the similarity between a web application and a HanPhone voice application. In both cases, content from the content server is rendered by a component for the end user. In the web application scenario, the browser on the user's computer renders the content. In the HanPhone voice application scenario, the HanPhone server renders it.

3. Creating content in HTML

Content rendered by HanPhone Server can be written in the native HanPhoneXML format or a trimmed down HTML format. HanPhone Server automatically converts the latter format into the former in real-time. An HTML document is assumed to be comprised of a block of content followed by a series of selections (hyperlinks) or followed by a form submission, as shown in Figure 3 below:

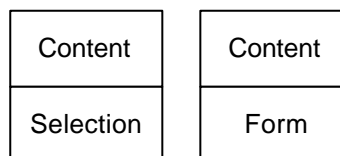


Figure 3. HTML supported by HanPhone

Any text after the <body> tag and before the first <a> or <form> tag is considered the content block of the HTML. If this block is followed by an <a> tag, all the <a> tags starting from this one are interpreted as a series of choices. When HanPhone renders this document, the content block is read before the choices, each assigned a key number, are read one by one in document order. The user can key in a number to jump to the URL represented by that number. Any text after the content block not enclosed by <a> tags is ignored.

If the content block is followed by a `<form>` tag, the user is prompted to enter a value for each `<input>` field defined in the form. After all the fields are filled, the form is automatically submitted to the URL defined in the action attribute in the `<form>` tag. Any text and tag after the first `</form>` tag is ignored.

Most of the other HTML tags such as formatting tags (``, `<u>`, etc.) and table tags (`<table>`, `<td>`, etc.) are ignored by HanPhone, although developers can make use of them so that they can better visualize the document in a web browser.

4. Creating content in HanPhoneXML

HTML might not be sufficient when a lower level of control over the rendering of the content is desired or when phone-specific actions must be performed. For example, actions like call-transfer, recording of spoken input and playback of pre-recorded content have no equivalent tags in HTML. HanPhoneXML directly exposes these actions to the developer so that they can invoke them easily.

HanPhoneXML is similar to VoiceXML in many ways (for more details, see *Interoperability between HanPhoneXML and VoiceXML*). However, its simpler design means that coding in HanPhoneXML is easier and more analogous to coding HTML. The following is an example of a HanPhoneXML document that presents a greeting and prompts the user to select a choice:

```
<?xml version="1.0"?>
<xml>
  <card>
    <content>Welcome to this example page.</content>
    <select default = "1" timeout="5" retry="2">
      <option key="1" href="news.xml">News</option>
      <option key="2" href="weather.xml">Weather</option>
    </select>
  </card>
</xml>
```

The user will hear the following:

Welcome to this example page. Press one for News. Press two for Weather.

The timeout attribute of the `<select>` tag tells HanPhone Server that the document at the

relative URL "news.xml" should be retrieved and rendered if the user does not select a option within five seconds. The retry attribute ensures that if the user presses neither one nor two, he/she will be given two more chances to make a valid choice before HanPhone Server disconnects the call and terminates the session.

5. Sessions, users and database connectivity

As mentioned in section 2, the logic of a HanPhone voice application resides on the content server. HanPhone Server only acts as an interpreter to render the content served from the content server via HTTP. Therefore, session control, keeping track of users and retrieval of data from a database are done on the content server using server-side programming languages such as PHP and ASP, JSP or even Java Servlet. HanPhone Server has a built-in HTTP/HTTPS client that supports cookie and therefore session control features provided by these programming languages can work with HanPhone Server too.

6. Voice recognition and other speech-related software

At the moment, voice recognition for Chinese languages is still not robust enough and DTMF detection is the primary way of gathering input from the user. However, HanPhone Server is designed to meet the ever-evolving business and technology demands. Its architecture allows integration of different voice recognition software in addition to speech synthesis software.

The HanPhoneXML language will be extended to expose any new functionality offered by HanPhone Server while maintaining backward compatibility. The majority of HanPhoneXML documents can be converted to VoiceXML.

7. Stability and Maintenance

HanPhone Server takes advantage of the robustness of Java and Dialogic voice board to provide a stable and economical voice application platform on Windows 2000 servers. Unlike traditional hardware-based IVRS, little knowledge in telephony is required of the supporting staff. Updating parts of a voice application is as simple as updating text in a HTML file on the content server. The HanPhone server needs not to be shut down during the process.

HanPhone Server can be configured to log all incoming calls in a database for further analysis. System administrator can also retrieve information about the server (such as checking the number of active sessions) and send administrative instructions to the server (such as starting and stopping it) using a client application/applet.