

## APPLICATION ABSTRACTION WITH DIALOG PURPOSE

### BACKGROUND OF THE INVENTION

[0001] The present invention generally relates to encoding computers to perform a specific application. More particularly, the present invention relates to controls for defining an application to perform recognition and/or audible prompting such as a server that generates client side markup enabled with recognition and/or audible prompting.

[0002] Small computing devices such as personal digital assistants (PDA), devices and portable phones are used with ever increasing frequency by people in their day-to-day activities. With the increase in processing power now available for microprocessors used to run these devices, the functionality of these devices are increasing, and in some cases, merging. For instance, many portable phones now can be used to access and browse the Internet as well as can be used to store personal information such as addresses, phone numbers and the like.

[0003] In view that these computing devices are being used for browsing the Internet, or are used in other server/client architectures, it is therefore necessary to enter information into the computing device. Unfortunately, due to the desire to keep these devices as small as possible in order that they are easily carried, conventional keyboards having all the letters of the alphabet as isolated buttons are usually not possible due to the limited surface area available on the housings of the computing devices.

[0004] To address this problem, there has been increased interest and adoption of using voice or speech to access information over a wide area network such as the Internet. For example, voice portals such as through the use of VoiceXML (voice extensible markup language) have been advanced to allow Internet content to be accessed using only a telephone. In this architecture, a document server (for example, a web server) processes requests from a client through a VoiceXML interpreter. The web server can produce VoiceXML documents in reply, which are processed by the VoiceXML interpreter and rendered audibly to the user. Using voice commands through voice recognition, the user can navigate the web.

[0005] Generally, there are two techniques of "speech enabling" information or web content. In the first technique, existing visual markup language pages typically visually rendered by a device having a display are interpreted and rendered aurally. However, this approach often yields poor results because pages meant for visual interaction usually do not have enough information to create a sensible aural dialog automatically. In addition, voice interaction is prone to error, especially over noisy channels such as a telephone. Without visual or other forms of persistent feedback, navigation through the web server application can be extremely difficult for the user. This approach thus requires mechanisms such as help messages, which are also rendered audibly to the user in order to help them navigate through the website. The mechanisms are commonly referred to as "voice dialogs", which also must address errors when incorrect information or no information is provided by the user, for example, in response to an audible question. Since the mechanisms are not commonly based on the visual content of the web page,

they cannot be generated automatically, and therefore typically require extensive development time by the application developer.

[0006] A second approach to speech enabling web content, includes writing specific voice pages in a new language. An advantage of this approach is that the speech-enabled page contains all the mechanisms needed for aural dialog such as repairs and navigational help. However, a significant disadvantage is that the application pages must then be adapted to include the application logic as found in the visual content pages. In other words, the application logic of the visual content pages must be rewritten in the form of the speech-enabled language. Even when this process can be automated by the use of tools creating visual and aural pages from the same specification, maintenance of the visual and speech enabled pages is usually difficult to synchronize. In addition, this approach does not easily allow multimodal applications, for example where both visual and speech interaction is provided on the web page. Since the visual and speech-enabled pages are unrelated, the input and output logic is not easily coordinated to work with each other.

[0007] To date, speech interaction is also cumbersome due to the organization or format currently used as the interface. Generally, the speech interface either tends to be tied too closely to the business logic of the application, which inhibits re-use of the elements of the speech interface in other applications, or the speech interface is too restricted by a simplistic dialog model (e.g. forms and fields).

[0008] There is thus an ongoing need to improve upon the architecture and methods used to provide speech recognition in an application such as server/client architecture such as the Internet. In particular, a method, system or authoring tool that addresses one, several or all of the foregoing disadvantages and thus provides generation of speech-enabled recognition and/or speech-enabled prompting in an application is needed.

### SUMMARY OF THE INVENTION

[0009] Controls are provided for a web server to generate client side markups that include recognition and/or audible prompting. The controls comprise elements of a dialog such as a question, answer, confirmation, command or statement. A module forms a dialog by making use of the information carried in the controls.

[0010] Each of the controls perform a role in the dialog. For instance, a statement control is used to generate corresponding markup for the client device to present information to the user, while a question control generates markups for the client device to ask a question. An answer control generates markup for the client device so that a grammar used for recognition is associated with an input field related to a question that has been asked. If it is unclear whether or not a recognized result is correct, a confirmation control can be activated and generate markup to confirm a recognized result. A command control generates markup that allows the user to provide commands, which are other than the expected answers to a specific question, and thus, allows the user to navigate through the web server application, for example. Each of the controls includes an attribute to indicate whether the associated control is available for activation. A module, when executed such as on a client,