Forms

A form in a VoiceXML document presents information and gathers input from the user. A form is represented by the <form> tag and has an ID attribute associated with it. The ID attribute is the name of the form.
Following is an example of a form element:

```
<form id="hello">
   <block>
   Hello world!
   </block>
</form>
```

In this example, the name of the form is "hello" and "Hello world!" is presented to the user.

Form Items

There are two types of form items, field items and control items. A field item prompts the user on what to say or key in, and then collects that information and fills in the field item variable. A field item also has grammars that define the allowed inputs, event handlers to process the resulting events, and a <filled> element that defines an action to be taken after the field item variable has been filled.

Following is a list of the types of field items:

< field >	The < field > form is the value of the field item obtained from the user via speech or DTMF.
< record >	The < record > form is the value of the field item which is an audio clip recorded by the user, such as a voice mail message, which can be collected by the <record> element.
< transfer >	The < transfer > form is used for transferring the user to another telephone number.
< object >	The < object > form invokes platform-specific objects with one or more properties.
< subdialog >	Like a function call, the form invokes a call to another dialog on the current page or another VoiceXML document. The task of a control item is to help control the gathering of the form's fields.

Following are two types of control items:

< block >	The < block > control is a sequence of statements used for prompting and computation.
< initial >	The < initial > control is useful in mixed initiative dialogs that prompt the user for information.

Variables and Conditions

A form item variable is associated with each form. The form item 'variable by default' is initially set to 'undefined' and contains a result (collected from the user) once a form item has been interpreted. You can define the name of a form item variable by using the name attribute. A guard condition exists for each form item. The guard condition tests whether the item's variable currently has a value. If a value exists, then the form item is skipped.

From:

Permanent link:

https://wiki.voximal.com/doku.php?id=legacy:vxi_developer_guide:voicexml_fundamentals:forms

Last update: **2017/07/28 23:53**