-continued

```
            city.
            </prompt >
        </Question>
        <Answer id="AnsDepCity" . . . />
    </Speech:QA>
```

[0089] In the example below, different prompts can be called depending on an internal condition of the question control **308**A. The ability to specify conditional tests on the prompts inside a question control **308**A means that changes in wording can be accommodated within the same functional unit of the question control **308**A.

```
    <Speech:QA        id="QADepCity"
            controlsToSpeechEnable="txtDepCity"
            runat="server" >
        <Question id="Q1" Answers="AnsDepCity" >
            <prompt count="1">
                Now I need to get the departure city.
                Where would you like to fly from?
            </prompt>
            <prompt count="2">
                Which departure city?
            </prompt>
        </Question>
        <Answer id="AnsDepCity" . . . />
    </Speech:QA>
```

[0090] Conditional QA Control

[0091] The following example illustrates how to determine whether or not to activate a QA control based upon information known to the application. The example is a portion of a survey application. The survey is gathering information from employees regarding the mode of transportation they use to get to work.

[0092] The portion of the survey first asks whether or not the user rides the bus to work. If the answer is:

[0093] Yes, the next question asks how many days last week the users rode the bus.

[0094] No, the "number of days rode the bus" question is bypassed.

```
<asp:Label id="lblDisplay1"
        text="Do you ride the bus to work?"
        runat="server"/ >
<asp:DropDownList id="1stRodeBusYN" runat="server">
    <asp:ListItem
selected="true">No</asp:ListItem>
    <asp:ListItem>Yes</asp:ListItem>
</asp:DropDownList >
<Speech:QA id="QA_RideBus
        ControlsToSpeechEnable="1stRodeBusYN"
        runat="server" >
    <SDN:Question id="Q_RideBus" >
        <prompt bargeIn="False">
        Do you ride the bus to work?
        </prompt>
    </SDN:Question>
    <SDN:Answer id="A_RideBus"
                autobind="False "
                StartEvent="onMouseDown"
```

-continued

```
                StopEvent="onMouseUp"
                runat="server"
    onClientReco="ProcessRideBusAnswer"
        <grammar src=". . ." /> <--! "yes/no"
grammar -->
    </SDN:Answer>
</Speech:QA>
<asp Label id="lblDisplay2"
        enabled="False"
        text="How many days last week did you ride
    the bus to work?"
        runat="server"/>
<asp:DropDownList id="1stDaysRodeBus" enabled="False"
runat ="server">
    <asp:ListItem selected="true"
>0</asp:ListItem>
        <asp:ListItem>1</asp:ListItem>
        <asp:ListItem>2</asp:ListItem>
        <asp:ListItem>3</asp:ListItem>
        <asp:ListItem>4</asp:ListItem>
        <asp:ListItem>5</asp:ListItem>
        <asp:ListItem>6</asp:ListItem>
        <asp:ListItem>7</asp:ListItem>
    </asp:DropDownList>
<Speech:QA id="QA_DaysRodeBus"
        ControlsToSpeechEnable="1stDaysRodeBus"
        ClientTest="RideBusCheck"
        runat="server" >
    <Question id="Q_DaysRodeBus" >
    <prompt bargeIn="False">
        How many days last week did you ride the
bus to work?
        </prompt>
    </SDN:Question>
    <SDN:Answer id="A_DaysRodeBus"
                autobind="False"
                StartEvent="onMouseDown"
                StopEvent ="onMouseUp"
                runat="server"
    onClientReco="ProcessDaysRodeBusAnswer"
        <grammar src=". . ." />     <--! "numbers"
grammar -->
    </SDN:Answer>
</Speech:QA>
<script language="jscript">
    function ProcessRideBusAnswer ( ) {
        <--! using SML attribute of the Event object,
determine yes or no answer -->
        <--! then select the appropriate item in the
dropdown listbox -->
        <--! and enable the next label and dropdown
listbox if answer is "yes" -->
        if <--! Answer is "yes" --> {
            1stRodeBusYN.selectedIndex=2
            lblDisplay2.enabled="true"
            1stDaysRodeBus.enabled="true" }
    {
    function RideBusCheck ( ) {
        if 1stRodeBusYN.selectedIndex="1" <--!
this is no -->
            then return "False"
        endif
    }
    function ProcessDaysRodeBusAnswer () {
        <--! case statement to select proper
dropdown item -->
    }
</script>
```

[0095] In the example provided above, the QA control "QA_DaysRodeBus" is executed based on a boolean parameter "ClientTest", which in this example, is set based on the function RideBusCheck( ) If the function returns a false condition, the QA control is not activated, whereas if a true