# Hotevents

Hotevents can only be produced through the Java API because they involve the use of the Unified CVP Voice Foundation Classes (VFCs), which are Java-only (see Voice Foundation Classes for more on the VFCs).

When a new hotevent is added to the workspace in Builder for Call Studio, the full name of a Java class must be entered in the hotevent dialog box. This class, when executed, is expected to produce the VoiceXML to run when the event is triggered. The VoiceXML generated by this class is placed in the root document automatically generated by VXML Server. Since the root document is cached by the voice browser, this class is executed only *once* per call, it is *not* called when the event is triggered and therefore does not have access to the Session API to obtain dynamic session information like the ANI, element and session data.

Following the standard design of the Java API, the hotevent class must implement a Java interface named `HoteventInterface` found in the `com.audium.server.proxy` package. The interface defines a single method `addEventBody` that is called when the call's root document is being generated. The method receives two VFC classes as arguments, a `VPreference` object and a `VEvent` object. The VoiceXML code to execute when the hotevent is triggered must be added to the `VEvent` object and the `VPreference` object is used to instantiate the VFC classes defining that VoiceXML. The method does not need to return anything as all content is encapsulated within the `VEvent` object passed by reference to the method.