

Using HotEvents to Gracefully Handle VoiceXML Error Events

Document ID: 91830

Introduction

Prerequisites

Components Used

Conventions

Background Information

NetPro Discussion Forums – Featured Conversations

Related Information

Introduction

This document describes how some VoiceXML error events can be handled gracefully with HotEvent elements instead of a hang-up on the caller.

Prerequisites

Components Used

The information in this document is based on Cisco Unified Call Studio, Universal Edition.

Conventions

Refer to Cisco Technical Tips Conventions for more information on document conventions.

Background Information

Symptoms: The call-flow designer wants to account for the more common VoiceXML error events and handle them in the call flow rather than allow default-error handling to occur.

Resolution: The HotEvent element listens for a particular event as specified in its Element Configuration. When that event occurs, its only exit state is followed, and the call flow can continue. While catching some events, such as a hang-up, is discouraged because it can affect the normal functionality of Cisco Unified Call Studio, Universal Edition, there are several events that you can handle in your call flow to improve the experience of the caller in error situations. Refer to your voice-browser documentation for a list of events that the browser can throw within a call.

Here is an example of how you can gracefully handle an automatic server restart (ASR) server if it goes down:

1. Configure a HotEvent to listen for the event that your voice browser throws in this situation; it can be something like resource.unavailable.asr.
2. Exit from the HotEvent to a Cisco Unified Call Studio, Universal Edition, element that explains to the caller that a minor error has occurred but that they can continue their call.
3. Connect the exit state of the Cisco Unified Call Studio, Universal Edition, element to an Application Transfer element.

4. Use the Application Transfer element to send the caller to a dtmf-only version of the application.

With this approach, if the ASR server goes down, the caller can continue the call. Dependent upon how caller input is stored, the caller possibly needs to reenter some data or go back in the call flow, but at least the caller can continue the interactive voice response (IVR) experience without the need to call back later.

Another example of this usage is for error.badfetch, which can occur if a media server goes down. In that case, you can use a HotEvent to route to a custom Action element that changes the default path to refer to a backup media server instead.

NetPro Discussion Forums – Featured Conversations

Networking Professionals Connection is a forum for networking professionals to share questions, suggestions, and information about networking solutions, products, and technologies. The featured links are some of the most recent conversations available in this technology.

NetPro Discussion Forums – Featured Conversations for Voice
Service Providers: Voice over IP
Voice & Video: Voice over IP
Voice & Video: IP Telephony
Voice & Video: IP Phone Services for End Users
Voice & Video: Unified Communications
Voice & Video: IP Phone Services for Developers
Voice & Video: General

Related Information

- [Technical Support & Documentation – Cisco Systems](#)

All contents are Copyright © 2006–2007 Cisco Systems, Inc. All rights reserved. Important Notices and Privacy Statement.

Updated: Jul 05, 2007

Document ID: 91830