



Call Control Requirements in a Voice Browser Framework

W3C Working Draft 13 April 2001

This version:

<http://www.w3.org/TR/2001/WD-call-control-reqs-20010413/>

Latest version:

<http://www.w3.org/TR/call-control-reqs/>

Previous versions:

(this is the first published version)

Editor:

Brad Porter, Tellme

Copyright © 2001 W3C® (MIT, INRIA, Keio), All Rights Reserved. W3C [liability](#), [trademark](#), [document use](#), and [software licensing](#) rules apply.

Abstract

This document describes requirements for mechanisms that enable fine-grained control of speech (signal processing) resources and telephony resources in a VoiceXML telephony platform. The scope of these language features is for controlling resources in a platform on the network edge, not for building network-based call processing applications in a telephone switching system, or for controlling an entire telecom network.

Status of this Document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. The latest status of this document series is maintained at the W3C.

This document describes the requirements for markup used for call control, as a precursor to work on a specification. You are encouraged to subscribe to the public discussion list <www-voice@w3.org> and to mail us your comments. To subscribe, send an email to <<mailto:www-voice-request@w3.org>> with the word *subscribe* in the subject line (include the word *unsubscribe* if you want to unsubscribe). A [public archive](#) is available online.

This document has been produced as part of the [W3C Voice Browser Activity](#), following the procedures set out for the [W3C Process](#). The authors of this document are members of the [Voice Browser Working Group \(W3C Members only\)](#).

Publication as a Working Draft does not imply endorsement by the W3C Membership. This is a draft document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to cite W3C Working Drafts as other than "work in progress". A list of current W3C Recommendations and other technical documents can be found at <http://www.w3.org/TR/>.

Table of Contents

1. [Introduction](#)
2. [Call Initiation Requirements](#)

3. [Inter-Session Communication Requirements](#)
4. [Inter-Session Communication Requirements](#)
5. [Conferencing Capabilities Requirements](#)
6. [Call Leg Management Requirements](#)
7. [Use Case Scenarios](#)
8. [Acknowledgements](#)

1. Introduction

The main goal of this subgroup is to establish a prioritized list of requirements for call control in a voice browser environment.

The process will consist of the following steps:

1. Collect requirements on call control.
2. Prioritize these requirements.
3. Distribute requirements to, and take feedback from, relevant groups working on call control in telephony systems.
4. Define specifications for call control components, based on the feedback received.

1.1 Scope

The core activity focuses on enabling extended call control functionality in a voice browser which supports telephony capabilities. The VoiceXML specification states that "VoiceXML is designed for creating audio dialogs that feature synthesized speech, digitized audio, recognition of spoken and DTMF key input, recording of spoken input, telephony, and mixed-initiative conversations." This activity will therefore specify richer telephony functionality in a voice browser framework.

The task is constrained to defining elements and capabilities which either provide augmented functionality to be used in combination with VoiceXML or enhance the existing functionality in VoiceXML.

This document specifies requirements that define the capabilities of a voice browser which supports telephony applications.

1.2 Interaction with Other Sub Groups

The activities of the Call Control Subgroup will be coordinated with the activities of the Dialog Subgroup (both of which are part of the W3C Voice Browser working group).

2. Call Initiation Requirements

This section deals with general requirements around accepting or placing a call. VoiceXML already specifies a simple behavior whereby calls to a particular phone number are answered and VoiceXML is immediately interpreted.

The call control system should be able to:

1. use a standard addressing scheme for telephony devices.
By standard addressing scheme this means a portable and extensible addressing mechanism capable of addressing current and future telephony devices.
2. place an outbound call.
This requirement does not specify who can place an outbound call or when it can happen, but presumably an outbound call can be initiated from any context if appropriate access is granted.
3. conditionally answer a call.
The implication is that a decision can be made before the system goes off-hook. This may mean by using information provided by the network when the call is presented (inbound call number, caller identification information, and so forth) or based on a user interaction for instance in the case of call waiting.

(Nice to have) Fax delivery falls into a class of outbound communication such as email, messaging, or other mechanisms. Inbound fax falls into a class of inbound communication protocols which might also include modems. We may find it advantageous to develop a system which can interact effectively with all of these communication mechanisms, though this is currently listed as "nice to have".

3. Interpreter Context Management Requirements

Computer-human interaction is handled by VoiceXML. In order to provide a richer human-computer experience with a sophisticated telephony network, certain content management techniques are required.

The call control system should be able to:

1. invoke a VoiceXML interpreter context associated with a call leg.
In the case of an inbound call, this is currently the behavior of VoiceXML. This requirement implies that capability should also be possible with an outbound call.
2. invoke and be able to terminate a separate VoiceXML dialog on a particular call leg
This requirement deals with the ability to interrupt a caller and present them with a new question or dialog asynchronously from the normal flow of conversation, for instance to notify the user of a new incoming call. The details of how this might be done are intentionally left out of the requirement definition.
3. place an outbound call from a VoiceXML interpreter context after original call leg terminates
(Nice to have) In certain cases you may want to continue the interpreter session with a different party after a disconnect. This is considered "nice to have" and may have substantial security implications if any user state is associated with that session.
4. suspend/resume an active VoiceXML dialog on a particular call leg
Suspension of an active dialog may be necessary to allow the caller to immediately deal with an interrupt event, but the caller may wish to later continue in that dialog.

4. Inter-Session Communication Requirements

Communication mechanisms are necessary to support a distributed network of telephony devices interacting together to provide advanced functionality. This section describes the basic requirements for inter-session communication.

The call control system should be able to:

1. send/receive asynchronous events to other VoiceXML sessions on different call legs
2. send/receive asynchronous events to an external system
3. send/receive synchronous events to other VoiceXML sessions on different call legs
4. send/receive synchronous events to an external system
5. standard inter-session communication protocol
This requirement addresses the expectation that communication will need to occur between disparate systems, thus requiring a standardized inter-session communication protocol. HTTP to dialog servers is one possible mechanism.
6. start and manage timers
This may be a useful capability for VoiceXML in general. For the purposes of this requirements document, assume we just need to be able to do this for call control, but may devise a mechanism that we can suggest generally for VoiceXML.
7. handle asynchronous events without having to interrupt a human-computer dialog or other operation in progress
Intent of this requirement is to allow for background processing of events without the end user's awareness.
8. handle asynchronous events and interrupt
9. initiate a different human-computer dialog based on an asynchronous event

5. Conferencing Capabilities Requirements

Conferencing multiple lines together is a specific area of functionality currently missing from VoiceXML. Two line discussions are allowed with the <transfer> tag, but the solution is not easily generalized to multi-party conferences. VoiceXML allows for only very minimal human-computer interaction during a transfer, leaving most of the dialog capabilities unavailable while two parties are connected. Ideally, human-computer interaction scripted through VoiceXML can be used to control multi-party conferences. This section describes principle requirements necessary to generalize for multi-party conferences for a voice browser environment.

The call control system should be able to:

1. create a conference call

This requirement simply specifies that there needs to be a way for a caller to conceptually initiate a new conference call. The requirement does not address the issues of managing conferencing resources, nor the underlying mechanism by which this conference is created. Extensive management of conferencing resources may or may not be beyond the scope of the call control task. Conceptually, however, a voice browser supporting call control mechanisms should allow a caller to initiate a conference call scenario.

2. join a conference call

This requirement specifies that conceptually a caller on a call leg should be able to join an active conference call. This requirement does not imply the mechanism by which a caller joins a conference or how the conference is first established.

3. control access rights to conference functionality

4. exit a conference call without call disconnect

This requirement allows the caller to continue interacting with a human-computer interface after leaving a multi-party conference

5. toggle speak only, mute, and moderator

6. each call leg can be on hold individually

7. VoiceXML dialog can whisper to and listen for hotwords from the caller on that particular call leg

8. conference both inbound & outbound audio channels from multiple call legs

9. VoiceXML dialog can act as a computer participant in a full conference (play audio and listen)

6. Call Leg Management Requirements

The core of telephony control in a voice browser involves managing call legs and audio streams. VoiceXML currently provides minimal or no capabilities for effectively managing calls legs or audio streams. This section describes some of the requirements needed for managing those call legs and audio streams.

The call control system should be able to:

1. create, control, and manage multiple calls legs

2. redirect a call leg

3. perform blind and consultation based transfer of call legs

4. have DTMF and speech grammars active on a leg even when the leg is bridged

5. control when to connect voice resource in transfer

6. control outbound audio on both legs of a transfer

7. a party can interact with a VoiceXML session while on hold

7. Use Case Scenarios

These use cases illustrate services that might be enabled by the combination of new telephony capabilities with a voice browser platform. These are not an exhaustive list, nor do these use cases imply that supporting these applications is a requirement. Instead these should be used to provide tangible context for discussing the requirements above.

These cases were generated based on significant input and examples provided by the subgroup

7.1 Call Center Customer Support Interactions

Acme customer support line wants to run a customer information and support service which allows users to call in, interact with an automated menu system using DTMF and voice. When the customer reaches a menu which requires an operator, the customer is placed in a hold queue for an available operator.

Alternatively, if the customer requests an operator at any point Acme would like to allow the customer to either wait for an operator, or continue navigating the system while in the hold queue. If the customer continues interacting with the automated system while waiting, Acme would like to be able to interrupt periodically with status about the hold queue and offer the customer the option of cancelling their request if their question has been answered by the automated system. When an operator is available, the customer's interactions are stopped and the operator is connected.

For training purposes, Acme would also like to be able to have a trainer listening when the customer is connected to the operator. This trainer could interrupt and provide hints to the new operator about how to answer the question. The customer would not be able to hear these hints.

7.2 Notification Services

Joe Edwards logs in to the Acme auction web site and registers that he wants to be notified if any pinball games come up for auction. He registers his cell phone number with the Acme auction web site. Later that day a pinball game becomes available. The auction site then contacts Joe. After a short advertisement, Joe can interact with an automated system using DTMF or voice to place a bid. At the same time, Joe can request to be notified by phone if he is outbid.

7.3 Company-wide Announcements

Acme has many distributed offices. Consequently, company-wide presentations are best done over the phone. During the call, only one speaker is allowed at a time. A single moderator controls which caller is active. At various points in the company-wide presentation the presenter would like to play pre-recorded customer testimonials.

7.4 Multi-party Conferencing

Because many of Acme's business groups work in different locations, multi-party conferencing is important for day-to-day operations. The conference can be initiated such that participants call in, or in a manner in which each participant is called directly.

During the conference, an individual participant can choose to be interrupted with status information (such as "new mail has arrived") or with call waiting. The conference participant can then decide whether to take action on the information or continue in the conference. After the action is complete, the participant can rejoin the conference.

7.5 Personal Assistant

Acme's sales force is dependent on the ability to get in touch with customers quickly and to always be available. Specifically, they need to be able to have their primary work number automatically redirected to any phone. They also need to use voice dialing to transfer to their customers. Each department has a budget for the amount of time they can use for transferred calls which needs to be updated based on usage.

8. Acknowledgements

The editor wishes to thank the members of the Call Control lexicon subgroup of the Voice Browser working group:

- RJ Auburn, Voxeo
- Dan Burnett, Nuance Communications
- Mike Cafarella, Tellme Networks
- Debbie Dahl, Unisys
- Pete Danielsen, Lucent
- James Ferrans, Motorola
- Warren Gallagher, Nuance Communications
- Gadi Inon, Comverse
- Don Jackson, Tellme Networks
- Sylvain Jodoin, Locus Dialog
- Gerald Karam, AT&T
- David Ladd, Dynamicsoft
- Bruce Lucas, IBM
- Scott McGlashan, PipeBeach
- Mitsuru Oshima, General Magic
- Brad Porter, Tellme Networks
- Ken Rehor, Nuance
- Jim Seidman, Verascape
- Saravanan Shanmugham, Cisco
- Liang Shen, VoiceGenie
- Pramod Sharma, Telera
- Enrico Silterra, Nortel Networks
- Corey Stohs, Cisco
- Jonathan Taylor, Voxeo
- Michael Tel, Openwave
- Luc Van Tichelen, Lernout & Hauspie
- Jenny Yao, Cisco