

# ECMAScript (JavaScript) in Your Voice Applications

Document ID: 91720

---

**Introduction**

**Prerequisites**

Components Used

Conventions

**Background Information**

**NetPro Discussion Forums – Featured Conversations**

**Related Information**

---

## Introduction

This document describes several reasons to include ECMAScript in the root document of your voice application for use at runtime.

## Prerequisites

## Components Used

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

## Conventions

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

## Background Information

**Symptoms:** The developer wonders if they should use ECMAScript for runtime processing.

**Resolution:** JavaScript (also called ECMAScript) that you include in the root document of your voice application is executed by the voice browser. This functionality is useful if you have a need for something to be done on the browser side, such as proprietary browser functionality, which is only available through JavaScript, or the acquisition of browser-centric information, such as the timestamp of the browser. (Note that Cisco Unified Call Services has achieved this latter example for quite some time.)

Any JavaScript in the root document of your voice application is only executed when it is referenced. There is no concept of a "main" method in the root document that can be executed unless it is called. Instead, it is used to put various utility methods in the root document for later access from within the VoiceXML of your application.

One of the strengths of Cisco Unified Call Services is the fact that it leverages a web application architecture and stores session data on the application server. This design reduces the burdens of processing a call and data storage off the voice browser and distributes it to the application server, but, since the session of the caller is stored on the application server (which includes element, session, call data, and more), that information is not available to JavaScript.

As a best practice, it is generally not recommended to use your voice browser to perform CPU-intensive processing or to store large amounts of caller-centric data since it could adversely affect other calls (which need CPU and memory resources for many other purposes). On the other hand, Cisco Unified Call Services is meant to do the heavy-lifting and pack the results up into VoiceXML, which the voice browser can then execute.

Information about how to add JavaScript to the root document of your application can be found in the Cisco Unified Call Studio online help (**Help > Help Contents** under *Root Document Settings*).

## 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: Aug 06, 2007

Document ID: 91720