This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

GVP HSG Pages

Traffic and Capacity Testing

5/11/2025

Traffic and Capacity Testing

Overview

Use this section to determine the required capacity of your GVP servers, based on anticipated traffic characteristics or by running tests on an existing system.

When measuring peak capacity on a single GVP machine, CPU usage is usually the determining factor—memory has not been an issue in most test cases. Therefore, the sample test results in this section concentrate on CPU usage and other criteria.

In addition, the Media Resource Control Protocol (MRCP) server that supports Automatic Speech Recognition (ASR) applications, must not share a host with a GVP server. You can use multiple MRCP servers for a particular test, however, it is important that the MRCP resources do not cause a bottleneck during testing.

This section contains test summary tables to assist in the difficult task of sizing in the face of so much raw data contained by the tables in the following sections. Each table is prefaced with a description of its intent, with suggestions for interpreting and applying the data.

The complexity of VoiceXML and CCXML applications impacts capacity testing, therefore, the Genesys QA performance testing results in this section are derived from test cases using four different VoiceXML applications and two different CCXML applications.

VoiceXML Application Profiles

VoiceXML performance testing was conducted on four major application profiles. Their characteristics are outlined in the tables below. The call flow duration for each application profile is for a single call or CD1 (see Call Duration (CD) and Peak Capacity (PC)).

Profile: VoiceXML_App1	**Profile: VoiceXML_App2**
A simple DTMF-only application designed to refill calling cards. • Total number of digits (DTMF input only) = 52, including:	A complex application designed for insurance coverage inquiries. • Speech input, including: ◦ Type of request

<table>
<tr><th>Profile: VoiceXML_App1</th><th>Profile: VoiceXML_App2</th></tr>
<tr><td>

Toll free number from the back of the card
Refill card PIN number
Refill dollar amount
Credit card number
Credit card expiration date
Zip Code of caller

Number of VoiceXML pages = 18
VoiceXML complexity = low
Number of audio prompts = 9
Number of audio files used in prompts (no TTS) = 107
ECMA script complexity = moderate
Number of VoiceXML pages = 6
Number of Java script help functions in each VoiceXML page = 13
Call flow duration:

74 seconds (pre GVP 8.1.2)
76 seconds (GVP 8.1.2 and higher)

</td><td>

ID card number
Confirmation
Relationship with insurance plan holder
Date of birth confirmation

Number of VoiceXML pages = 10
VoiceXML complexity (~ 1 MB of content) = High
Number of audio prompts = 7
Number of audio files used in prompts (3 with TTS) = 29
ECMA script complexity = high
Call flow duration:

70 seconds (ASR engine)
55 seconds (ASR engine simulator)

</td></tr>
</table>

<table>
<tr><th>Profile: VoiceXML_App3</th><th>Profile: VoiceXML_App4</th></tr>
<tr><td>
QA ASR/TTS load application.

Speech input, including:

Words
Digits
Hotkey (NGI)
Yes or no confirmation

Number of VoiceXML pages = 1
VoiceXML complexity = low
Number of audio prompts = 7 prompts involve 7 audio files and 7 TTS
ECMA script complexity = low
Call flow duration = 62 seconds

</td><td>
Composer-generated application designed for IVR-assisted banking.

Input a total of 20 digits (DTMF only):

Input current customer number
Confirm contact ID
Input debit menu option
Input debit banking menu
Input personal option
Input 6 digit secure code

Number of VoiceXML pages = 20
VoiceXML complexity = medium (~ 400 KB of content)
Number of audio prompts = 6 (no TTS, 12 audio files)

</td></tr>
</table>

Profile: VoiceXML_App3	Profile: VoiceXML_App4
	• ECMA script complexity = moderate (4 general JavaScript function files) • Call duration = 85 seconds

Profile: VoiceXML_App5	Profile: VoiceXML_App6
VoiceXML_App1 with IVR recording function. In addition to running the VoiceXML_App1 application, IVR recording was also started when the VoiceXML_App1 began and the call was recorded until the end. **Recording details** • No of channels = 2 • Recording type = mp3 • Bit rate = 16 kbps • Recording destination = http • Recording metadata = enabled	Simple IVR recording application with continuous speech input from the caller. • Number of VoiceXML pages = 1 • VoiceXML complexity = low • Number of audio prompts = 2 (2 audio files) • Call flow duration = 75 seconds (NGI) **Recording details** • No of channels = 2 • Recording type = mp3 • Bit rate = 16 kbps • Recording destination = http • Recording metadata = enabled

Profile: VoiceXML_App7	Profile: VoiceXML_App8

A simple voice input application designed to get transcript from Google ASR directly from MCP (NativeGSR).
• Number of VoiceXML pages = 1
• VoiceXML complexity = low
• Number of audio prompts = 5
• Call flow duration:
    • ~ 5.5 seconds
	A simple voice input application designed to use Text to Speech service from Google directly from MCP (NativeGTTS).
	• Number of VoiceXML pages = 1
	• VoiceXML complexity = low
	• Number of TTS prompts = 1
	• Number of characters in TTS prompt = 344
	• Call flow duration:
	    • ~ 22.5 seconds

CCXML Application Profiles

CallControlXML (CCXML) performance testing was conducted on two major application profiles. Their characteristics are outlined below. The call flow duration for each application profile is for a single call or CD1 (see Call Duration (CD) and Peak Capacity (PC)).

Profile: CCXML_App1	Profile: CCXML_App2

An outbound application that joins multiple call legs, dialogs, and conferences.
 	
• Includes the following steps:
  ◦ Call customer and connect to a dialog
  ◦ Call agent and connect to dialog
  ◦ Exit agent dialog
  ◦ Exit customer dialog
  ◦ Create conference
  ◦ Join customer and agent to conference
  ◦ Disconnect agent
  ◦ Disconnect customer
  ◦ Destroy conference
• Number of CCXML (JSP) pages = 2
• CCXML complexity = medium
• Customer call duration = 8.7 seconds
• Agent call duration = 8.6 seconds
• Conference call duration = 6 seconds
	Simple conference recording call.
	
	• Includes the following steps:
	  ◦ Create a call to agent
	  ◦ Agent receives an invite and a dialog is created for agent to ring back
	  ◦ Agent answers the call and a conference is created to join caller and agent
	  ◦ Conference is established and dialog is created for recording
	  ◦ Call is disconnected from caller after 15 seconds of recording
	• Number of CCXML pages = 1
	• Number of VoiceXML pages = 2
	• CCXML complexity = medium
	• Call duration = 21 seconds

top | toc