Sizing

Overview

An important consideration in sizing a contact center is to determine the worst-case contact center profile for the number of calls that are in each state. For instance, if you observe the contact center at its busiest instant in the busiest hour, observe how many calls you find in the following states:

- **Self-service**—Calls that are executing applications using the VXML Server.
- **Queue and collect**—Calls that are in queue for an agent or are executing prompt-and-collect type self-service applications.
- **Talking**—Calls that are connected to agents or to third-party TDM VRU applications.

In counting the number of calls that are in the talking state, count only calls that are using Unified CVP or gateway resources. To determine whether a call in talking state is using resources, consider how the call gets transferred to that VRU or agent. If the call is transferred using VoIP, it continues to use an Ingress VoiceXML Gateway port and it continues to use a Unified CVP resource, because Unified CVP continues to monitor the call and provides the ability to retrieve it and redeliver it at a later time. Unified CVP also continues to monitor calls that are sent to a TDM target, using both an incoming and an outgoing TDM port on the same gateway or on a different gateway (that is, toll bypass). Calls that are transferred to VRUs or agents in this manner are counted as talking calls.

However, if the call was transferred through *8 TNT, hookflash, Two B Channel Transfer (TBCT), or an ICM NIC, neither the gateway nor Unified CVP play any role in the call. Both components have reclaimed their resources, therefore, such calls are not counted as talking calls.

Include in the overall call counts those calls that have been transferred back into Unified CVP for queuing or self-service, using either blind or warm methods. For example, if a warm transfer is used and the agent is queued at Unified CVP during the post-route phase, the call will use two ports due to two separate call control

sessions at Unified CVP. Because these calls usually do not contribute to more than 5 percent or 10 percent of the overall call volume, you can easily overlook them.

The definitions of these call states differ from the definitions used for port licensing purposes. Similarly, the call state determination does not influence with whether the agents are Unified CCE agents or ACD agents, nor does it matter whether the customer intends to use Unified CVP to retrieve and redeliver the call to another agent or back into self-service.

Note You should size the solution for the number of ports in use for calls in a talking state to agents. Even though licenses for those ports do not have to be purchased when using Unified CCE agents, TDM agents do require a Call Director license.

In addition to the overall snapshot profile of calls in the contact center, you must also consider the busiest period call arrival rate in terms of calls per second. You need this information for the contact center as a whole because it is difficult to identify the exact maximum arrival rate. You can use statistical means to arrive at this number, except in very small implementations, which is seldom the critical factor in determining sizing.

You can begin sizing each component in the network following the information in the overview section. The overview section, deals with the number and type of physical components required to support the Unified CVP system, but it does not include any discussion of redundancy. For an understanding of how to extend these numbers to support higher reliability, see Unified CVP Design for High Availability.

Note In Unified CVP, the Call Server, VXML Server, and Media Server are combined as one installation as CVP Server. Installing the CVP Server installs all three components. In the earlier versions, Call Server, VXML Server, and Media Server could be installed on different machines.

Call Server Sizing

Note Call Server is not used in Model #1 (Standalone Self-Service). This section does not apply to these deployments.

Unified CVP Call Servers are sized according to the number of calls they can handle, in addition to their maximum call arrival rate.

Table 1: Call Server Call Rate

Call Server	Capacity
Maximum SIP Calls	900
Sustained Calls per Second (SIP)	10

Note For UCS performance numbers, see the *Virtualization for Cisco Unified Customer Voice Portal* page at http://docwiki.cisco.com/wiki/Virtualization_for_Cisco_Unified_Customer_Voice_Portal.

Example

Consider the following Call Server calculations:

Each Call Server can handle 900 SIP calls. Each Call Server is further limited to a sustained call arrival rate of 10 call per second (cps) for SIP. However, Model #4 is exempt from this limitation because the Call Server in that model does not perform any SIP processing.

Specifically, the number of Call Servers required is the higher of these totals:

((Self Service) + (Queue and Collect) + Talking) / 900, rounded up

or

(Average call arrival rate) / 10, rounded up.

In addition, calls delivered to the Cisco Unified Communications Manager cluster should be load balanced among the subscribers in the cluster and should not exceed 2 calls per second (cps) per subscriber.

Call Server Log Directory Size Estimate

When you use the Agent Greeting feature, performance of Unified CVP Call Servers is reduced by 25 percent. This reduction occurs because the servers operate at 75 percent of calls per second (CPS) of a system that is not using the Agent Greeting feature.

Size your system using the methods detailed in the guide, then multiply the CPS by 75 percent:

- For example, 10 CPS on a UCS platform without Agent Greeting translates into 7.5 CPS on a Call Server with Agent Greeting enabled.
- Ports required are calculated based on the CPS and duration of agent greeting, and must be determined from the total supported ports of a server.

Call Server Log Directory Size Estimate

Use the following formula to calculate the estimated space per day (in gigabytes) for the Call Server Directory log file:

3.5 * R

R equals the number of calls per second.

For proper serviceability, reserve enough space to retain from five to seven days of log messages.

To set the log directory size, in the Operations Console, choose the Infrastructure tab for Call Server set up.

VXML Server Sizing

VXML Server call rate calculations are described in the table and examples in this section.

One VXML Server can handle up to 900 calls. If you are using VXML Servers, size them according to the following formula:

Calls / 900, rounded up,

Calls refers to the number of calls that are in VXML Server self-service applications at that snapshot in time.

Note For UCS performance numbers, go to this location http://docwiki.cisco.com/wiki/Virtualization_for_Unified_CVP.

You can configure Unified CVP to use HTTPS on the VXML Server and on the Unified CVP IVR Service (IVR Service can generate basic VoiceXML documents and is part of the Call Server). Due to the large processing overhead of HTTPS, the Tomcat application server can achieve a maximum of 275 simultaneous connections, depending on the configuration.

Set Cache Command

- IOS VoiceXML Gateway—Configure the Cisco IOS VoiceXML Gateway with the HTTPS option. If this configuration setting is not available then this can severely impact the performance and sizing of the VXML Gateway and the overall solution in general with HTTPS.

  http client connection persistent

  http client cache memory pool 15000

  http client cache memory file 1000

- Cisco VVB—Configure the Cisco VVB with the HTTPS option. If this configuration setting is not available then this can severely impact the performance and sizing of the VXML browser and the overall solution in general with HTTPS. The following CLI commands are used for modifying cache properties (for more details, see *Cisco Virtualized Voice Browser Operations Guide*):

  set vvb cache browser_cache_size

  set vvb cache dom_cache_capacity

  set vvb cache enable_browser_cache

  set vvb cache enable_browser_cache_trace

  set vvb cache enable_dom_cache

The following table provides simultaneous call information for HTTPS calls using various applications and call flow models.

Table 2: HTTPS Simultaneous Calls for Unified CVP Servers

Call Server Type, Application, and Call Flow Model	Number of Simultaneous Calls
VXML ServerMax simultaneous HTTPS connections with Tomcat (Standalone Call Flow model)	275
Call Server and VXML ServerMax simultaneous HTTPS connections with Tomcat (Comprehensive Call Flow model)	275

Note In all of the above scenarios, the Reporting and Datafeed options are disabled. Also, Cisco IOS Release 12.4(15)T5 or later release is required on the gateway to support the HTTPS option. Mainline Cisco IOS is not supported.

IOS VoiceXML Gateway

VoiceXML Gateway Sizing for Agent Greeting

The additional VoiceXML Gateway ports required are calculated based on calls per second (CPS) and the duration of the agent greeting. The agent greeting is counted as one additional call to the VoiceXML Gateway.

Use the following formula to determine the additional ports required for the Agent Greeting feature:

```
Total ports = Inbound ports + [( Agent Greeting Duration / Total call duration) *
Inbound ports]
```

For example, if you estimate 120 calls, each with a 60-second call duration, you have 2 CPS and a requirement of 120 inbound ports. If you assume that the agent greeting duration is 5 seconds on every call, then the overall calls per second is 4 CPS, but the number of ports required is 130.

```
Total Ports = 120 inbound ports + [(5-second agent greeting duration / 60-second total call duration) *
120 inbound ports] = 130 total ports.
```

VoiceXML Gateway Agent Greeting Prompt Cache Sizing

For sizing agent greeting prompt cache, consider the following example:

The following calculation shows that a 1-minute long file in the g711uLaw codec uses approximately 1/2 MB:

```
64 kbits/sec = 8 kbytes/sec (bit rate for g711uLaw codec)
8 kb/sec * 60 seconds - 480 kb (~ 0.5 MB
```

- The maximum memory used for prompt cache in a Cisco IOS router is 100 MB and the maximum size of a single file should be 600 KB.
- Number of Agent Greetings cached using the above sizing numbers are provided here:
  - 5-second greeting—40 KB, that is, approximately 25 greetings per MB. This typical use case scenario provides caching for approximately 80*25 agent = 2000 agent greetings with 80 percent space reserved for Agent Greeting.
  - 60-second greeting—480 KB, that is, approximately 2 greeting per MB. The worst case scenario provides caching for approximately 50*2 agent = 100 agents with 50 percent space used for Agent Greeting.

Note For Cisco VVB, maximum cache size is 2 GB and the above calculation can be based on 2 GB instead of 100 MB.

Media Server Sizing for Agent Greeting

Media Server sizing usually is not provided due to the following reasons:

- Diverse requirements of a media server based on specific deployment requirements
- Wide range of hardware that is used for media servers

However, the following sizing profile is for a Media Server that is used with the Agent Greeting feature.

Example load:

- 700 agents
- 15-second greeting (118-kb greeting file)
- 30-minute content expiration

Media Server hardware equivalent to the following (or better) is required to handle the above profile:

- UCS platform with RAID 5 (media server only)
- UCS platform with RAID 5 (media server only)
- UCS platform with RAID 5 (collocated media/call server)

Unified CVP Coresidency

Self-service means that a call requires SIP call control and runs an application on the VXML Server. Queue and collect means that a call requires SIP call control and runs an application using Microapps only on the Call Server.

The following example applies for VoiceXML and HTTP sessions only. The same values apply to both coresident and distributed deployments of Call Servers and VXML Servers.

The number of servers required using SIP call control would be as follows:

((Self Service) + (Queue and Collect) + Talking) / 900, rounded up

((900) + (500) + 3700) / 900 = 6 servers

If you use the Cisco Unified Border Element as a Session Border Controller (SBC) for flow-through calls to handle VoiceXML requirements, then you must use the sizing information provided in the example. The Cisco Unified Border Element is limited to the maximum number of simultaneous VoiceXML sessions or calls as outlined provided in the example for the particular situation and hardware platform.

If you use the Cisco Unified Border Element as a Session Border Controller (SBC) to handle flow-through calls only (no VoiceXML), then consider Voice Activity Detection (VAD) and see the sizing information in

the *Cisco Unified Border Element Ordering Guide*, available at http://www.cisco.com/en/US/prod/collateral/voicesw/ps6790/gatecont/ps5640/order_guide_c07_462222.html.

Coresident Unified CVP Reporting Server and Unified CVP Call Server

Reporting Server can be coresident with the Call Server, but only for Standalone VoiceXML deployments. A Call Server usually is not needed in a Standalone VoiceXML deployment, but if reporting is desired, a Call Server is required to send the reporting data from the VXML Server to the Reporting Server. When the Reporting Server is coresident with a Call Server, the Call Server is not processing any SIP calls, but is relaying reporting data from the VXML Server.

The coresident Call Server does not have a significant impact on performance in this model, therefore, the sizing information in the Reporting Server section does not change.

Note If Unified Border Element is to be used as a Session Border Controller (SBC), use these procedures:

- To handle flow-through or flow-around calls only (no VXML), including VAD, use the Unified Border Element Ordering Guide for sizing.
- For flow-through or flow-around calls and with VXML requirements, use the sizing information in the *Design Guide for Cisco Unified Customer Voice Portal*. Unified Border Element is limited to the maximum number of simultaneous VXML sessions and calls as outlined for the particular situation and hardware platform.

Cisco Unified SIP Proxy

Unified CVP supports only the Cisco Unified SIP Proxy (CUSP) Server.

Information on CUSP architecture, feature, configuration, and data sheets is available at http://wwwin.cisco.com/artg/products/voice_video/cusp/.

The CUSP baseline tests are done in isolation on the proxy, and capacity numbers (450 to 500 cps) should be used as the highest benchmark. In a Unified CVP deployment, a CUSP proxy sees incoming calls from the TDM Gateway, from Unified CVP, and from the UCM SIP trunk. With a SIP back-to-back user agent in CVP, the initial call setup from the proxy, involves an inbound call immediately followed by an outbound call (whether for IVR or to ACD). Later in the call, CVP may transfer the call to an agent, which involves an outbound leg, and reinvites to the inbound leg. A ringtone service setup is also available which also involves a separate outbound call and a reinvite to the caller. Reinvites on the caller leg occur at CVP transfer or during supplementary services.

Note If the Proxy Server Record Route is set to on, it impacts the performance of the Proxy Server (as shown in the CUSP baseline matrix) and it also breaks the high availability model because the proxy becomes a single point of failure for the call. Always turn the Record Route setting of the proxy server to off to avoid a single point of failure, to allow fault tolerance routing, and to increase the performance of the Proxy Server.

A CVP call from the Proxy Server, involves four separate SIP calls on an average:

- Caller inbound leg

- VXML outbound leg
- Ringtone outbound leg
- Agent outbound leg

The standard for Unified CVP and CUSP proxy sizing is to define four SIP calls for every one CVP call, so the CPS rate is 500 / 4 = 125. The overall number of active calls is a function of Call Rate (CPS) * call handle time (CHT). Assuming an average call center call duration of 180 seconds (CHT), you get an overall active call value of 22,500 calls. Because one Call Server can handle approximately 900 simultaneous calls, it allows a single CUSP proxy to handle the load of 18 CVP Call Servers. A customer deployment should include consideration of the CPS and the CHT to size the proxy for their solution.

The standard for Unified CVP and CUSP proxy sizing is to define four SIP calls for every one CVP call, so the CPS rate is 500 / 4 = 125. The overall number of active calls is a function of Call Rate (CPS) * call handle time (CHT). Assuming an average call center call duration of 180 seconds (CHT), you get an overall active calls value of 22,500 calls. Because one Call Server can handle approximately 900 simultaneous calls, it allows a single CUSP proxy to handle the load of 18 CVP Call Servers. A customer deployment should include consideration of the CPS and the CHT to size the proxy for their solution.

Unified CVP Video Service

Cisco Unified CVP release 7.0 introduced capabilities for video-capable agents of Cisco Unified Contact Center Enterprise (Unified CCE).

The same Unified CVP Call Server can be used to service both video calls and traditional audio calls as long as the audio calls are handled using the Unified CVP comprehensive call flow. If any model other than the Comprehensive Model is used for the audio calls, then separate Call Servers must be used for the video and audio calls.

Basic Video Service Sizing

The Unified CVP Basic Video Service uses the comprehensive call flow, and it requires Call Server, VXML Server, and IOS VoiceXML Gateways. Sizing of these components for the Basic Video Service is done in the same manner as for traditional audio applications.

Cisco Unified Video conferencing hardware, Radvision IVP, and Radvision iContact are not required for the Basic Video Service.

Note Video call is not supported in Cisco VVB.

Reporting Server Sizing

There are many variables to consider for sizing the Reporting Server. Different VoiceXML applications have different characteristics, and those characteristics influence the amount of reporting data generated. Some of these factors are:

- The types of elements used in the application

- The granularity of data required
- The call flow that the users take through the application
- The length of calls
- The number of calls

To size the Reporting Server, you must first estimate how much reporting data is generated by your VoiceXML application. The example applications and the tables in subsequent sections of this chapter help you to determine the number of reporting messages generated for your application.

Once you have determined the number of reporting messages generated by your application, complete the following steps for each VoiceXML application:

1. Estimate the calls per second that the application receives.
2. Estimate the number of reporting messages for your application.

Use the following equation to determine the number of reporting messages generated per second for each VoiceXML application:

A# = %CPS * CPS * MSG

Where:

- A# is the number of estimated reporting messages per second for an application. Complete one calculation per application (A1, A2, …, An).
- CPS is the number of calls per second.
- %CPS is the percentage of calls that use this VoiceXML application.
- MSG is the number of reporting messages this application generates. To determine the number of reporting messages generated by your application, use the information provided in the sections on Reporting Message Details and Example Applications, on page 12.

Next, estimate the total number of reporting messages that your deployment generates per second by adding the values obtained from the previous calculation for each application:

A(total) = A1+ A2+…..+An

This is the total number of reporting messages generated per second by your VoiceXML applications. Each Reporting Servers can handle 420 messages per second. If the total number of reporting messages per second for your deployment is less than 420, you can use a single Reporting Server. If the number is greater, you need to use multiple Reporting Servers and partition the VoiceXML applications to use specific Reporting Servers.

Multiple Reporting Servers

If the number of messages per second (as determined in steps 1 and 2 in the previous section) exceeds the Reporting Server capacity, then the deployment must be partitioned vertically.

When vertically partitioning to load balance reporting data, a Unified CVP system designer must consider the following requirements that apply to deployments of multiple Reporting Servers:

- Each Call Server and VXML Server can be associated with only one Reporting Server.

- Reports cannot span multiple Informix databases.

For more information on these requirements, see the *Reporting Guide for Cisco Unified Customer Voice Portal* available at:

http://www.cisco.com/en/US/products/sw/custcosw/ps1006/products_installation_and_configuration_guides_list.html

When designing Unified CVP deployments with multiple Reporting Servers, observe the following guidelines:

- Subdivide applications that generate more combined call processing and application messages than are supported by one Reporting Server.
- VoiceXML can be filtered, and filtering out noninteresting data creates more usable data repositories that support higher message volume.
- Configure the dial plan and other available means to direct the incoming calls to the appropriate Call Server and VXML Server.

If you need to combine data from multiple databases, you can use these possible options:

- Exporting reporting data to Excel, comma-separated values (CSV) files, or another format that allows data to be combined outside of the database
- Exporting reporting data to CSV files and importing it into a customer-supplied database
- Extracting data to a customer-supplied data warehouse and running reports against that data

Reporting Message Details

The following table lists the various elements or activities and the number of reporting messages generated by them.

Table 3: Number of Reporting Messages per Element or Activity

Element or Activity	Number of Reporting Messages (Unfiltered)
Start	2
End	2
Subflow Call	2
Subflow Start	2
Subflow Return	2
Throw	2
Alert	2
Subdialog_start	2
Subdialog_return	2
Hotlink	2

Element or Activity	Number of Reporting Messages (Unfiltered)
HotEvent	2
Transfer w/o Audio	2
Currency w/o Audio	2
Flag	2
Action	2
Decision	2
Application Transfer	2
VoiceXML Error	2
CallICMInfo (per call)	2
Session Variable (per change)	2
Custom Log (per item)	2
Play (Audio file or TTS)	2
LeaveQueue	2
Callback_Disconnect_Caller	3
Callback_Add	4
Callback_Get_Status	4
Callback_Set_Queue_Defaults	4
Callback_Update_Status	4
Callback_Enter_Queue	5
Callback_Reconnect	5
Get Input (DTMF)	5
Callback_Validate	6
Get Input (ASR)	9
Form	10
Digit_with_confirm	20
Currency_with_confirm	20
ReqICMLabel	30

Note These elements are required in every application and cannot be filtered.

Example Applications

This section presents some examples of applications to estimate the number of reporting messages that are generated by your particular application.

Low Complexity

Total: 16 reporting messages per call.

Table 4: Example: Applications with Low Complexity

Element Type	Approximate Number of Reporting Messages
Start	2
Subdialog_start	2
Play element	2
Play element	2
Play element	2
Play element	2
Subdialog_end	2
End	2

Medium Complexity DTMF Only

Total: 51 reporting messages per call.

Table 5: Example: Applications with Medium Complexity Using ASR

Element Type	Approximate Number of Reporting Messages
Start	2
Subdialog_start	2
Play element	2
Get input	9
Play element	2
Get input	9

Element Type	Approximate Number of Reporting Messages
Form	10
Input	9
Transfer with audio	2
Subdialog_end	2
End	2

Medium Complexity Using Automatic Speech Recognition

Total: 39 reporting messages per call.

Table 6: Example: Applications with Medium Complexity DTMF Only

Element Type	Approximate Number of Reporting Messages
Start	2
Subdialog_start	2
Play element	2
Get input	5
Play element	2
Get input	5
Form	10
Input	5
Transfer with audio	2
Subdialog_end	2
End	2

High Complexity Using Automatic Speech Recognition

Total: 107 reporting messages per call.

Table 7: Example: Applications with High Complexity Using ASR

Element Type	Approximate Number of Reporting Messages
Start	2
Subdialog_start	2
Icmrequrestlabel	30

Element Type	Approximate Number of Reporting Messages
Form	10
ASR capture	9
Digit with confirm	20
Form	10
Digit with confirm	20
Subdialog_end	2
End	2