Cisco Unified Customer Voice Portal (CVP) Solution Reference Network Design (SRND)

Cisco Unified Customer Voice Portal (CVP) Release 7.x
August 8, 2008

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: OL-15989-02
# CONTENTS

Preface xi  
Audience xi  
Revision History xi  
Obtaining Documentation, Obtaining Support, and Security Guidelines xii  

## CHAPTER 1  
Unified CVP Architecture Overview 1-1  
What is VoiceXML? 1-1  
What is the Cisco Unified Customer Voice Portal? 1-2  
Unified CVP Components 1-3  
- Unified CVP Server 1-4  
- Unified CVP VoiceXML Server 1-4  
- Unified CVP VoiceXML Studio 1-5  
- Unified CVP Reporting Server 1-5  
- Unified CVP Operations Console Server 1-6  
- Unified CVP Video Media Server 1-6  
- Cisco Ingress Voice Gateway 1-6  
- Cisco VoiceXML Gateway 1-7  
- Cisco Egress Gateway 1-7  
- Multipoint Conference Unit 1-7  
- Video Endpoints 1-8  
- Cisco Unified Communications Manager 1-8  
- Cisco Unified Contact Center 1-9  
- Cisco Gatekeeper 1-9  
- SIP Proxy Server 1-9  
- DNS Server 1-10  
- Radvision IVP Server 1-11  
- Content Services Switch 1-11  
- Third-Party Media Server 1-12  
- Third-Party Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) Servers 1-12  

Call Flows 1-13  
- Typical SIP Unified CVP Call Flow 1-13  
- Typical H.323 Unified CVP Call Flow 1-14  

Design Process 1-14  
- Choosing SIP or H.323 1-14
Functional Deployment Models  2-1
Standalone VoiceXML Server  2-1
  Protocol-Level Call Flow  2-2
  Transfers and Subsequent Call Control  2-2
Call Director  2-3
  SIP Protocol-Level Call Flow  2-4
  H.323 Protocol-Level Call Flow  2-5
  Transfers and Subsequent Call Control  2-5
Comprehensive  2-6
  SIP Protocol-Level Call Flow  2-7
  H.323 Protocol-Level Call Flow  2-8
  Transfers and Subsequent Call Control  2-10
VRU Only  2-10
  Protocol-Level Call Flow  2-11
Basic Video Service  2-13
Full Video Service  2-13
  Protocol-Level Call Flow  2-15

Distributed Deployments  3-1
Distributed Gateways  3-1
  Ingress and/or Egress Gateway at the Branch  3-1
  Ingress or VoiceXML Gateway at Branch  3-1
  Co-located VoiceXML Servers and Gateways  3-2
  Gateways at the Branch, with Centralized VoiceXML Servers  3-3
  Cisco Unified Communications Manager  3-3
    Unified CM as an Egress Gateway  3-3
    Unified CM as an Ingress Gateway  3-3
Call Survivability in Distributed Deployments  3-4
Call Admission Control Considerations  3-4
  Gatekeeper Call Admission Control  3-5
  Unified CM Call Admission Control  3-5
    H.323 Call Flows  3-5
    Multiple Cisco Unified CM Clusters  3-7
Designing Unified CVP for High Availability  4-1

Overview  4-1

Layer 2 Switch  4-3

Originating Gateway  4-4
  Configuration  4-4
  Call Disposition  4-5

SIP Proxy  4-5
  Configuration  4-6
    SIP Proxy Server Configuration  4-6
    Cisco IOS Gateway Configuration  4-6
  Call Disposition  4-8

Unified CVP SIP Service  4-8
  Configuration  4-9
  Configuring High Availability for Calls in Progress  4-9
  Call Disposition  4-10

Gatekeeper  4-11
  Gatekeeper Redundancy Using HSRP  4-11
  Gatekeeper Redundancy Using Alternate Gatekeeper  4-12
  Configuration  4-12
    HSRP Configuration  4-12
    Alternate Gatekeeper  4-13
  Call Disposition  4-13

Unified CVP H.323 Service  4-14
  Configuration  4-14
    Configuring High Availability for New Calls  4-14
    Configuring High Availability for Calls in Progress  4-15
    Additional Cisco IOS Gateway Configuration  4-16
  Call Disposition  4-16

Unified CVP IVR Service  4-16
  Configuration  4-17
  Call Disposition  4-17

VoiceXML Gateway  4-17
  Configuration  4-18
    Centralized VoiceXML Gateways  4-18
    Distributed VoiceXML Gateways (Co-Resident Ingress Gateway and VoiceXML)  4-19
Contents

Distributed VoiceXML Gateways (Separate Ingress Gateway and VoiceXML) 4-19
H.323 Alternate Endpoints 4-22
Call Disposition 4-22
Hardware Configuration for High Availability on the Voice Gateways 4-22
Content Services Switch (CSS) 4-23
Configuration 4-23
Call Disposition 4-23
Media Server 4-24
Configuration When Using Unified CVP Microapplications 4-24
Call Disposition When Using Unified CVP Microapplications 4-25
Configuration When Using Unified CVP VoiceXML Studio Scripting 4-25
Unified CVP Video Media Server 4-25
Call Disposition 4-25
Unified CVP VoiceXML Server 4-26
Configuration 4-26
Standalone Self-Service Deployments 4-26
Deployments Using ICM 4-26
Call Disposition 4-26
Radvision IVP Server 4-27
Call Disposition 4-27
Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) Server 4-27
Configuration 4-27
Standalone Self-Service Deployments 4-28
Deployments Using ICM 4-28
Call Disposition 4-28
Cisco Unified Communications Manager 4-29
Configuration 4-29
Call Disposition 4-29
Intelligent Contact Management (ICM) 4-30
Configuration 4-30
Call Disposition 4-30

CHAPTER 5

Interactions with Cisco Unified ICM 5-1
Network VRU Types 5-1
Overview of Unified ICM Network VRUs 5-2
Unified CVP as a Type 10 VRU 5-3
Unified CVP as Type 5 VRU 5-3
Unified CVP as Type 3 or 7 VRU (Correlation ID Mechanism) 5-4
Unified CVP as Type 8 or 2 VRU (Translation Route ID Mechanism) 5-5
Network VRU Types and Unified CVP Deployment Models 5-5
- Model #1: Standalone Self-Service 5-7
- Model #2: Call Director 5-7
- Model #3a: Comprehensive Using ICM Micro-Apps 5-7
- Model #3b: Comprehensive Using CVP VoiceXML Server 5-7
- Model #4: VRU Only 5-8
  - Model #4a. VRU Only with NIC Controlled Routing 5-8
  - Model #4b. VRU Only with NIC Controlled Pre-Routing 5-8

Hosted Implementations 5-9
- Overview of Hosted Implementations 5-9
- Using Unified CVP in Hosted Environments 5-10
- Unified CVP Placement and Call Routing in a Hosted Environment 5-11
- Network VRU Type in a Hosted Environment 5-12

Deployment Models and Sizing Implications for Calls Originated by Cisco Unified Communications Manager and ACDs 5-13
- Using Third-Party VRUs 5-14

CHAPTER 6
Calls Originated by Cisco Unified Communications Manager 6-1
- Differences in Calls Originated by Cisco Unified Communications Manager 6-1
  - Customer Call Flows 6-2
    - Unified ICM Outbound Calls with Transfer to IVR 6-2
    - Internal Help Desk Calls 6-2
    - Warm Consultative Transfers and Conferences 6-2
  - Protocol Call Flows 6-3
    - Model #1: Standalone Self-Service 6-3
    - Model #2: Call Director 6-3
    - Model #3a: Comprehensive Using ICM Micro-Apps 6-5
    - Model #3b: Comprehensive Using CVP VoiceXML Server 6-6
  - Deployment Implications 6-6
    - Unified ICM Configuration 6-6
    - Hosted Implementations 6-7
    - Cisco Unified Communications Manager Configuration 6-7
    - Gatekeeper or SIP Proxy Dial-Plan Configuration 6-7
    - Sizing 6-7
      - Gateways 6-8
      - MTP Resources 6-8

CHAPTER 7
Gateway Options 7-1
- PSTN Gateway 7-1
VoiceXML Gateway with DTMF or ASR/TTS  7-2
VoiceXML and PSTN Gateway with DTMF or ASR/TTS  7-2
TDM Interfaces  7-2
Cisco Unified Border Element  7-3
Gateway Choices  7-4
Gateway Sizing  7-5
Using MGCP Gateways  7-7

**Design Implications for VoiceXML Server**  8-1
What is VoiceXML over HTTP?  8-1
Multi-Language Support  8-2
Differences in the Supported Web Application Servers  8-2
Where to Install Unified Call Studio  8-3

**Network Infrastructure Considerations**  9-1
Bandwidth Provisioning and QoS Considerations  9-1
Unified CVP Network Architecture Overview  9-2
Voice Traffic  9-2
Call Control Traffic  9-2
Data Traffic  9-4
Bandwidth Sizing  9-4
VoiceXML Documents  9-4
Media File Retrieval  9-5
H.323 Signaling  9-6
SIP Signaling  9-6
ASR and TTS  9-6
Voice Traffic  9-8
Call Admission Control  9-8
QoS Marking  9-9
Blocking Initial G.711 Media Burst  9-9
Network Security Using Firewalls  9-10

**Call Transfer Options**  10-1
Release Trunk Transfers  10-1
Takeback-and-Transfer (TNT)  10-2
Hookflash and Wink  10-2
Two B Channel Transfer (TBCT)  10-3
Contents

IX

Cisco Unified Customer Voice Portal (CVP) 7.x Solution Reference Network Design (SRND)

ICM Managed Transfers 10-4
SIP Refer Transfer 10-5
H.323 Refer Transfer 10-5
Intelligent Network (IN) Release Trunk Transfers 10-5
VoiceXML Transfers 10-6

CHAPTER 11
Using the GKTMP NIC 11-1
The Cisco Gatekeeper External Interface 11-1
The Unified ICM GKTMP NIC 11-1
Typical Applications of GKTMP with Unified CVP 11-2
Protocol-Level Call Flow 11-3
Deployment Implications 11-5

CHAPTER 12
Media File Options 12-1
Deployment and Ongoing Management 12-1
Co-Resident Call Server, Media Server, and VoiceXML Server 12-1
Bandwidth Calculation for Prompt Retrieval 12-3
Configuring Caching and Streaming in Cisco IOS 12-3
Streaming and Non-Streaming 12-3
Caching 12-3
Caching Query URLs 12-4
TCP Socket Persistence 12-4
Cache Aging 12-4
Branch Office Implications 12-6

CHAPTER 13
Managing, Monitoring, and Reporting 13-1
Operations Console: Managing and Monitoring 13-1
End-to-End Tracking of Individual Calls: Log Files 13-2
Formal Reporting 13-3
Backup and Restore 13-3
More Information 13-4

CHAPTER 14
Sizing 14-1
Sizing Overview 14-1
Unified CVP Call Server 14-2
Unified CVP VoiceXML Server 14-3
Unified CVP Co-Residency 14-3
Preface

This document provides design considerations and guidelines for deploying enterprise network solutions that include the Cisco Unified Customer Voice Portal (CVP).

This document builds upon ideas and concepts presented in the latest version of the Cisco Unified Contact Center Enterprise (CCE) Solution Reference Network Design (SRND), which is available online at

http://cisco.com/go/srnd

Note

Unless stated otherwise, the information in this document applies to Cisco Unified CVP 7.x (releases 7.0 and later).

Audience

This design guide is intended for the system architects, designers, engineers, and Cisco channel partners who want to apply best design practices for the Cisco Unified Customer Voice Portal (CVP).

This document assumes that you are already familiar with basic contact center terms and concepts and with the information presented in the Cisco Unified CCE SRND. To review those terms and concepts, refer to the documentation at the preceding URL.

Revision History

This document may be updated at any time without notice. You can obtain the latest version of this document online at

http://cisco.com/go/srnd

Visit this Cisco.com website periodically and check for documentation updates by comparing the revision date (on the front title page) of your copy with the revision date of the online document.

The following table lists the revision history for this document.

<table>
<thead>
<tr>
<th>Revision Date</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>August 8, 2008</td>
<td>Updates added for licensing and several other topics.</td>
</tr>
<tr>
<td>February 27, 2008</td>
<td>Initial release of this document for Cisco Unified CVP 7.0.</td>
</tr>
</tbody>
</table>
Obtaining Documentation, Obtaining Support, and Security Guidelines

For information on obtaining documentation, obtaining support, providing documentation feedback, security guidelines, and also recommended aliases and general Cisco documents, see the monthly What’s New in Cisco Product Documentation, which also lists all new and revised Cisco technical documentation, at:

Unified CVP Architecture Overview

Over the past two decades, many customers have invested in TDM-based interactive voice response (IVR) applications to automate simple customer transactions such as checking account or 401K account inquiries. In addition, many TDM-based IVR platforms were based on proprietary development environments and hardware platforms, which typically meant restricting the customer’s integration options with automatic speech recognition (ASR) and text-to-speech (TTS) solutions. Over the past few years there has been a dramatic shift to using VoiceXML standards-based technology to support the next generation of IVR applications.

What is VoiceXML?

Voice eXtensible Markup Language, or VoiceXML, is a markup language similar to HTML, that is used for developing IVR services and leverages the power of web development and content delivery. VoiceXML was designed for creating audio dialogs that feature synthesized speech, digitized audio, recognition of speech or dual-tone multifrequency (DTMF) key input, and recording of spoken input. It is a common language for content providers, tool providers, and platform providers, and it promotes service portability across implementation platforms. VoiceXML separates service logic from user interaction and presentation logic in VoiceXML voice web pages. It also shields application authors from low-level, platform-specific IVR and call control details. VoiceXML is easy to use for simple interactions, yet it provides language features to support complex IVR dialogues.

VoiceXML programs are rendered (or executed) by a VoiceXML browser, much like an HTML program is rendered via an internet browser (such as Internet Explorer). A Cisco Voice Gateway (or router) can provide the VoiceXML browser function. Specifically, the Cisco IOS VoiceXML Gateway provides the VoiceXML browser function. For small deployments, the Ingress Voice Gateway and VoiceXML Gateway are typically deployed in the same router.

In the most simple call processing scenario, upon arrival of a new call, the voice gateway dial peer matches the call to an available VoiceXML gateway port. The VoiceXML gateway port represents a Voice over IP (VoIP) endpoint and can be logically thought of as a voice response unit (VRU) port. Upon arrival of the new call, the VoiceXML gateway (that is, the VRU) sends an HTTP request to a VoiceXML server for instruction. The URL contained in the HTTP request correlates to a specific VoiceXML doc. In response to the HTTP request, the VoiceXML server sends the requested, dynamically generated VoiceXML doc to the VoiceXML gateway (that is, the voice browser) to be rendered. A typical VoiceXML doc is short and will prompt the caller for some input and then include the results in a new HTTP request that redirects the caller to another URL and VoiceXML doc. Because a typical call will require numerous prompts and caller inputs, there will be numerous VoiceXML documents that need to be rendered and an infinite number of possible paths through these VoiceXML documents.
What is the Cisco Unified Customer Voice Portal?

Unified CVP is both a product and a solution. There are specific software products that are delivered as part of the media kit for the Unified CVP product. However, the Unified CVP product relies upon other key components of the Unified CVP solution. From a solution perspective, Unified CVP is a VoiceXML-based solution that provides carrier-class IVR and IP switching services on Voice over IP (VoIP) networks. The Unified CVP feature set includes:

- **IP-based IVR services**
  Unified CVP can perform basic prompt-and-collect or advanced self-service applications with CRM database integration as well as ASR and TTS integrated via Media Resource Control Protocol (MRCP). Unified CVP supports both audio and video IVR services.

- **IP-based queuing treatment**
  Callers waiting for agents are presented personalized prompts and music and are prioritized based upon their CRM profile.

- **Integration with Cisco Unified Contact Center**
  Unified CVP integrates with Cisco Unified Contact Center via a VRU Peripheral Gateway (PG). This integration allows Cisco Unified Contact Center Enterprise (Unified CCE) to control Unified CVP VoIP switching and IVR services. This integration also allows Unified CCE to control the agent selection application and to initiate the Real-Time Transport Protocol (RTP) stream transfer from the VoiceXML gateway to the selected agent. Unified CVP integration with Unified CCE requires that the traditional Cisco Unified Communications Manager PG be used for Unified CCE integration with Cisco Unified Communications Manager.

Unified CCE can be integrated with Unified CVP via the Cisco Unified Intelligent Contact Manager (ICM) System PG and the parent-child deployment model. This integration method provides callers with some simple up-front menus and prompts by the parent Unified ICM and Unified CVP, and it intelligently routes the calls via skill groups to the best Cisco Unified Contact Center Express or Enterprise child. Queuing control and agent selection are handled by the child contact center solution. In this model, it is also easy for a TDM automatic call distributor (ACD) to play the role of a child. All call transfers between Unified CVP and children will retain call data, and the ICM will provide enterprise-wide browser-based consolidated reporting.

Unified CVP integration is not directly supported with the Unified CCE System PG (which is also used by System Unified CCE). The Unified CCE System PG supports only the Cisco Unified IP IVR. Unified CVP works only with System PG children via the parent-child deployment model. Unified CVP can also provide IVR services for Unified CCE outbound IVR campaigns and post-call customer surveys.
IP-based call switching
Unified CVP can route and transfer calls between voice gateways and IP endpoints. Voice gateways provide natural integration of TDM ACDs and PBXs with the PSTN. After completing the routing or transfer of a call, Unified CVP maintains H.323 or SIP call control to provide for subsequent call control. This call control enables Unified CVP to provide switching services similar to takeback-and-transfer (TNT) between IP endpoints via the Unified ICM Enterprise (ICME) interface. Integration with Cisco Unified Presence Server and a gatekeeper helps to provide easily managed dial plans.

Unified CVP Operations Console
Unified CVP has a single management portal for administrative and configuration tasks for both Unified CVP product components and portions of some Unified CVP solution components (such as gateways). Unified CVP also integrates with Cisco Contact Center Support Tools, but Support Tools must run on a separate physical server.

VRU reporting
Unified CVP provides a centralized reporting database for access to real-time and historical reporting.

**Unified CVP Components**

As mentioned previously, Unified CVP is both a product and a solution. The Cisco Unified Customer Voice Portal (CVP) product consists of the following components:

- Unified CVP Server, page 1-4
- Unified CVP VoiceXML Server, page 1-4
- Unified CVP VoiceXML Studio, page 1-5
- Unified CVP Reporting Server, page 1-5
- Unified CVP Operations Console Server, page 1-6

The following components of the Unified CVP solution are not part of the Unified CVP product but are still required for a complete solution:

- Cisco Ingress Voice Gateway, page 1-6
- Cisco VoiceXML Gateway, page 1-7
- Cisco Egress Gateway, page 1-7
- Cisco Unified Communications Manager, page 1-8
- Cisco Unified Contact Center, page 1-9
- Cisco Gatekeeper, page 1-9
- SIP Proxy Server, page 1-9
- DNS Server, page 1-10
- Content Services Switch, page 1-11
- Third-Party Media Server, page 1-12
- Third-Party Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) Servers, page 1-12

The following sections discuss each of these components in more detail. Depending on the particular deployment model you choose, some of the above components might not be required.
Unified CVP Server

The Unified CVP Server component provides the following independent services, which all run on the same Windows 2003 server:

- Unified CVP Server: SIP Service
  This service implements a SIP Back-to-Back User Agent (B2BUA). This B2BUA accepts SIP invites from ingress voice gateways and typically directs those new calls to an available VoiceXML gateway port. After completing call setup, the Unified CVP B2BUA acts as an active intermediary for any subsequent call control. While the Unified CVP SIP signaling is hairpinned through this service, this service does not touch the RTP traffic.

  Integrated into this B2BUA is the ability to interact with the Cisco Unified ICM via the ICM Service. This integration provides the ability for the SIP Service to query the Unified ICM for routing instruction and service control. This integration also allows Unified ICM to initiate subsequent call control to do things such as requesting that a caller be transferred from queue to an agent or transferred from one agent to another agent.

- Unified CVP Server: ICM Service
  This service is responsible for all communication between Unified CVP components and Unified ICM. It sends and receives messages on behalf of the SIP Service, the IVR Service, and the H.323 Service.

- Unified CVP Server: IVR Service
  This service creates the VoiceXML pages that implement the Unified CVP Microapplications based on Run VRU Script instructions received from Unified ICM. The IVR Service functions as the VRU leg (in Unified ICM Enterprise parlance), and calls must be transferred to it from the SIP Service in order to execute microapplications. The VoiceXML pages created by this module are sent to the VoiceXML gateway to be executed.

- Unified CVP Server: H.323 Service (Formerly known as the Unified CVP Voice Browser)
  This service interacts with the IVR Service to relay call arrival, release, and transfer call control between it and the other H.323 components. This service is needed only for deployments using H.323.

A Unified CVP Call Server can be deployed co-resident with the Unified CVP VoiceXML Server or Media Server.

For hardware details, refer to the latest version of the Hardware and System Software Specification for Cisco Unified CVP (formerly called the Bill of Materials), available at:


Unified CVP VoiceXML Server

This component executes complex IVR applications by exchanging VoiceXML pages with the VoiceXML gateway's built-in voice browser. Like almost all other Unified CVP product components, it runs within a Java 2 Enterprise Edition (J2EE) application server environment such as Tomcat or WebSphere, and many customers add their own custom-built or off-the-shelf J2EE components to interact with back-end hosts and services. VoiceXML Server applications are written using VoiceXML Studio and are deployed to the VoiceXML Server for execution. The applications are invoked on an as-needed basis by a special microapplication which must be executed from within the Unified ICME routing script.
VoiceXML Server can also be deployed in a Standalone configuration that does not include any Unified ICME components. In this model, applications are invoked as a direct result of calls arriving in the VoiceXML gateway, and a single post-application transfer is allowed.

The Unified CVP VoiceXML Server can be installed co-resident with a Unified CVP Call Server or the Media Server.

The Unified CVP VoiceXML Server can execute on Windows 2003 or AIX 5.3 servers. For hardware requirements and details, refer to the latest version of the Hardware and System Software Specification for Cisco Unified CVP (formerly called the Bill of Materials), available at


**Unified CVP VoiceXML Studio**

This component is the service creation environment (script editor) for VoiceXML Server applications. It is based on the open source Eclipse framework, and it provides advanced drag-and-drop graphical editing as well as the ability to insert vendor-supplied and custom-developed plug-ins that enable applications to interact with other services in the network. VoiceXML Studio is essentially an offline tool whose only interaction with the VoiceXML Server is to deliver compiled applications and plugged-in components for execution.

The Unified CVP VoiceXML Studio can execute on Windows 2003 or AIX 5.3 workstations or servers. Because the license is associated with the IP address of the machine on which it is running, customers typically designate one or more data center servers for that purpose. Unified CVP VoiceXML Studio cannot run on machines also running a headless version of the Cisco Security Agent.

For additional hardware details, refer to the latest version of the Hardware and System Software Specification for Cisco Unified CVP (formerly called the Bill of Materials), available at


**Unified CVP Reporting Server**

The Unified CVP Reporting Server is a Windows 2003 server that hosts an IBM Informix Dynamic Server (IDS) database management system. The Reporting Server provides consolidated historical reporting for a distributed self-service deployment. The database schema is prescribed by the Unified CVP product, but the schema is fully published so that customers can develop custom reports based on it. The Reporting Server receives reporting data from the IVR Service, the SIP Service (if used), and the VoiceXML Server. The Reporting Server depends on the Call Server to receive call records. For Standalone VoiceXML Server deployments, one Call Server is needed per Reporting Server. The Reporting Server must be local to the Call Server(s) and VXML Server(s) that it is servicing; deploying the Reporting Server at a remote location across the WAN is not supported. Multiple Reporting Servers should be used and placed at each site when Call Server(s) and VXML Server(s) exist at multiple locations.

The Reporting Server does not itself perform database administrative and maintenance activities such as backups or purging. However, Unified CVP provides access to such maintenance tasks through the Operations Console.
Unified CVP Operations Console Server

The Operations Console Server is a Windows 2003 server that provides an Operations Console for the browser-based administration and configuration for all Unified CVP product components, and it offers shortcuts into the administration and configuration interfaces of other Unified CVP solution components. This is a required component in all Unified CVP deployments and must run on a separate machine, except in lab deployments.

The Operations Console also offers a direct link to Support Tools, which can collect trace logs and perform other diagnostic and instrumentation functions on many solution components. The Operations Console is, in effect, the dashboard from which an entire Unified CVP deployment can be managed.

The Operations Console must itself be configured with a map of the deployed solution network. It can then collect and maintain configuration information from each deployed component. Both the network map and the configuration information are stored locally on the server, where it can be backed up by off-the-shelf backup tools. A web browser-based user interface, the Operations Console, provides the ability to both display and modify the network map and the stored configuration data and to distribute such modifications to the affected solution components.

The Operations Console can display two views of configuration parameters for managed components. The runtime view shows the status of all configuration parameters as those components are currently using them. The configured or offline view shows the status of all configuration parameters that are stored in the Operations Server database and will be deployed to the device the next time a Save and Deploy option is executed.

The Operations Console allows configuration parameters to be updated or preconfigured even when the target component is not online or running. If the target server (without its services) comes online, the user can apply the configured settings to that server. These settings will become active when that server’s services also come online. Only then will they be reflected in the runtime view.

The Operations Console Server is not a redundant component. As such, the Operations Console Server cannot be duplicated within a deployment. Backups of the configuration database should be taken regularly or whenever changes are made.

Unified CVP Video Media Server

The Video Media Server (VMS) is a Unified CVP component that provides live and recorded video content to callers with video-enabled devices. The VMS stores, manages, and streams 3GP videos and is accessible via two web interfaces: the Cisco Unified Customer Voice Portal Video Media Server Administration and the Cisco Unified Customer Voice Portal Video Desktop.

This component is required for Full Video Service deployments only.

Cisco Ingress Voice Gateway

The Cisco Ingress Voice Gateway is the point at which an incoming call enters the Unified CVP system. It terminates TDM calls on one side and implements VoIP on the other side. It serves as a pivot point for extension of calls from the TDM environment to VoIP endpoints. Therefore, WAN bandwidth is conserved because no hairpinning of the media stream occurs. Unified CVP Ingress Voice Gateways support both SIP and H.323. Media Gateway Control Protocol (MGCP) voice gateways are supported if they are registered with Cisco Unified Communications Manager.
Supported gateways include the Cisco 2800 Series, 3700 Series, 3800 Series, 5350XM, 5400 HPX, and 5400 XM. The Cisco AS5850 ERSC is also supported as an ingress voice gateway. For the most current list of supported gateways, refer to the latest version of the Hardware and System Software Specification for Cisco Unified CVP (formerly called the Bill of Materials), available at http://www.cisco.com/en/US/products/sw/custcosw/ps1006/prod_technical_reference_list.html

Cisco VoiceXML Gateway

The VoiceXML Gateway hosts the Cisco IOS Voice Browser. This component interprets VoiceXML pages from either the Unified CVP Server IVR Service or the VoiceXML Server. The VoiceXML Gateway encodes .wav files and accepts DTMF input. It then returns the results to the controlling application and waits for further instructions.

The Cisco VoiceXML Gateway can be deployed on the same router as the Unified CVP Ingress Voice Gateway. This model is typically desirable in deployments with small branch offices. But the VoiceXML Gateway can also run on a separate router platform, and this model is typically desirable in deployments with large or multiple voice gateways, where only a small percentage of the traffic is for Unified CVP. This model allows an organization to share PSTN trunks between normal office users and contact center agents and to route calls based upon the dialed number.

The Cisco VoiceXML Gateway can encode .wav files stored in flash memory or on a third-party media server. Prompts retrieved from a third-party media server can be cached in the router to reduce WAN bandwidth and prevent poor voice quality. The VoiceXML doc will provide a pointer to the location of the .wav file to be played or it will provide the address of a TTS server to generate a .wav file. The VoiceXML Gateway interacts with ASR and TTS servers via MRCP.


Cisco Egress Gateway

The Egress Voice Gateway is used only when calls need to be extended to TDM networks or equipment such as the PSTN or a TDM ACD. While the RTP stream goes between the ingress and egress voice gateway ports, the signaling stream logically goes through the Unified CVP Server and ICM in order to allow subsequent call control (such as transfers).

Multipoint Conference Unit

The Multipoint Conference Unit (MCU) consists of three components, the MCU Chassis, the MCU card, and the EMP card (sometimes referred to as the MVP card). The Cisco MCU Chassis, MCU card, and EMP/MVP card are available under the IPVC-3545-xxx Cisco part numbers.

Note

A firmware conversion is required to make the Cisco MCU and EMP/MVP cards compatible with the software components from Radvision, Inc., that are used in the Unified CVP Full Video Service. A firmware conversion utility for the MCU and EMP cards is provided by Radvision on the installation CD for their software components.
The MCU and EMP register to and communicate with the Radvision IVP Server. Multiple MCUs and EMPs can be used to provide scalability and redundancy and are pooled together for the IVP Server to utilize dynamically when needed throughout a call.

The MCU and EMP are required for Full Video Service deployments only.

**Video Endpoints**

When using the Unified CVP Basic Video Service, the following video endpoints are supported:

- Cisco Unified IP Phone 7985G
- Cisco Unified Video Advantage
- Cisco TelePresence

When using the Unified CVP Full Video Service, the following video endpoints are supported:

- Cisco Unified IP Phone 7985G
- Cisco Unified Video Advantage
- Radvision 3G Video Gateway and 3G Mobile Video Phone

**Cisco Unified Communications Manager**

Cisco Unified Communications Manager (Unified CM) is the main call processing component of a Cisco Unified Communications system. It manages and switches VoIP calls among IP phones. Unified CM combines with Cisco Unified Intelligent Contact Manager Enterprise (Unified ICME) to form Cisco Unified Contact Center Enterprise (Unified CCE). Unified CVP interacts with Unified CM primarily as a means for sending PSTN-originated calls to Unified CCE agents. SIP gateway calls are routed to an available Unified CM SIP trunk, and H.323 gateway calls are routed to an available Unified CM H.323 trunk.

The following common scenarios require calls to Unified CVP to originate from Unified CM endpoints:

- A typical office worker (not an agent) on an IP phone dials an internal help desk number.
- An agent initiates a consultative transfer that gets routed to a Unified CVP queue point.
- A Cisco Unified Outbound Dialer port transfers a live call to a Unified CVP port for an IVR campaign.

A single Unified CM can originate and receive calls from both SIP and H.323 devices. PSTN calls that arrived on MGCP voice gateways registered with Unified CM can also be routed or transferred to Unified CVP via either SIP or H.323, depending upon the deployment model chosen.

Unified CM is an optional component in the Unified CVP solution. Its use in the solution depends on the type of call center being deployed. Pure TDM-based call centers using ACDs, for example, typically do not use Unified CM (except when migrating to Cisco Unified CCE), nor do strictly self-service applications using the Unified CVP Standalone Self-Service deployment model. Unified CM generally is used as part of the Cisco Unified CCE solution, in which call center agents are part of an IP solution using Cisco IP Phones, or when migrating from TDM ACDs.
Only specific versions of Unified CM are compatible with Unified CVP solutions. Unified CVP is supported with SIP only if Cisco Unified CM 5.0 or later release is used. Unified CVP is supported with H.323 for Cisco Unified CM 4.x or later releases. For full details on version compatibility, consult the latest version of the Hardware and System Software Specification for Cisco Unified CVP (formerly called the Bill of Materials), available at http://www.cisco.com/en/US/products/sw/custcsw/ps1006/prod_technical_reference_list.html

Cisco Unified Contact Center

Either Cisco Unified CCE or Cisco Unified Intelligent Contact Management (ICM) is a required component when advanced call control (IP switching, transfers to agents, and so forth) is required in Unified CVP. The Hosted versions of these products might also be used for this purpose. Unified ICM provides call-center agent management capabilities and call scripting capabilities. Variable storage capability and database access through the Unified CCE or Unified ICM application gateways are also powerful tools. A Unified CVP application can take advantage of these capabilities because Unified CVP applications can be called from within a Unified CCE or Unified ICM script in a non-standalone Unified CVP deployment model.

The Unified CVP Call Server maintains a GED-125 Service Control Interface connection to Unified CCE or Unified ICM. GED-125 is a third-party-control protocol in which a single socket connection is used to control many telephone calls. From the point of view of Unified CCE or Unified ICM, the Unified CVP Call Server is a voice response unit (VRU) connected to Unified CCE or Unified ICM, just as all other GED-125 VRUs are connected. Unified CVP is simply a VRU peripheral to Unified CCE or Unified ICM.

Cisco Gatekeeper

The gatekeeper is a network element used by H.323 gateways for call routing. It is an optional component. However, most H.323 installations incorporate an H.323 gatekeeper for dial plan configuration and bandwidth management.

One scenario for gatekeeper usage with Unified CVP is to map specific dialed numbers to specific Unified CVP Servers or VoiceXML gateways. Another scenario for gatekeeper usage with Unified CVP is to load-balance new calls to a set of Unified CVP Servers or VoiceXML gateways. A third scenario for gatekeeper usage with Unified CVP is to route the transfer of callers from a VoiceXML gateway port to a Cisco IP Phone.

Two gatekeeper failover mechanisms are supported: Hot Standby Router Protocol (HSRP) and Alternate Gatekeepers.

For more information on H.323 gatekeepers, refer to the Cisco gatekeeper product documentation available online at Cisco.com.

SIP Proxy Server

The SIP Proxy Server is the component that routes individual SIP messages among SIP endpoints. It plays a key role in Unified CVP’s high-availability architecture for call switching. It is designed to support multiple SIP endpoints of various types and to implement load balancing and failover among these endpoints. Currently the only supported SIP Proxy Server is the Cisco Unified Presence Server, which has a built-in SIP Proxy function. Any other SIP Proxy Server must be tested by the customer with Unified CVP to ensure compatibility.
Chapter 1  Unified CVP Architecture Overview

Unified CVP Components

The Cisco Unified Presence Server SIP Proxy cannot use DNS SRV for outbound calls; it must be configured with multiple static routes in order to do load balancing and failover. The static routes can point to an IP address or a regular DNS A host record.

Unified CVP may also be deployed without a SIP Proxy Server. In such cases, some of the same functions can be provided by the Unified CVP Server SIP Service. If a SIP Proxy Server is not used, then Ingress Gateways and Unified CMs must point directly to Unified CVP. In such a deployment, load balancing is done via DNS SRV lookups from the gateway to the DNS Server. Load balancing of calls outbound from Unified CVP (outbound call leg) can be done in a similar fashion.

The following guidelines apply to the use of a Cisco Unified Presence server as a SIP Proxy:

- A Unified CM publisher is required in order to install Cisco Unified Presence. Therefore, you need at least one Unified CM publisher if you plan on using the Cisco Unified Presence server as a SIP Proxy (even for a TDM-only deployment with no Unified CM or Unified CCE agents). Unified CM does not need any Device License Units to perform this function.
- A single Cisco Unified CM 6.x publisher can support two Cisco Unified Presence servers; therefore, a single Unified CM 6.x cluster can have two Cisco Unified Presence servers. If you need more than two Cisco Unified Presence servers, additional Unified CM 6.x publishers are required. However, a Cisco Unified CM 7.x publisher can support six Cisco Unified Presence nodes per cluster, contained in three dual-node sub-clusters.
- The Cisco Unified Presence servers can be clustered over the WAN, provided the following conditions are met:
  - Clustering over the WAN with Cisco Unified Presence servers requires either Cisco Unified CM 6.x or 7.x.
  - The Cisco Unified Presence servers must not be used for presence or for registrations.
  - The Cisco Unified Presence servers can be used in SIP Proxy mode only.
  - The WAN round-trip delay between the Cisco Unified Presence server and Unified CM must not exceed 40 msec.

In situations where redundancy across two or more sites is required but clustering over the WAN with Cisco Unified Presence servers is not needed or cannot be supported, you will need at least one Unified CM publisher and one Cisco Unified Presence server at each site. Cisco Unified Presence configuration data is not shared between clusters, so you must configure each Cisco Unified Presence server with dial plan information.

- If you have multiple Cisco Unified Presence servers, in order for them to provide redundancy to Unified CVP, you must configure a DNS SRV record that provides load balancing and/or failover pointing at both servers. You then configure Unified CVP to use this single DNS SRV record as the SIP Proxy Server.
- If you have multiple Cisco Unified Communications Manager clusters, you do not need a Cisco Unified Presence server attached to each cluster for SIP Proxy functionality. It is possible for a single Cisco Unified Presence server to provide SIP Proxy services for multiple clusters. However, depending on where the clusters are located, you might want to have multiple Cisco Unified Presence servers for redundancy (such as with clustering over the WAN).

DNS Server

This optional component may be installed anywhere in the network. Its purpose in general is to resolve hostnames to IP addresses. However, it can also be configured with multiple IP addresses for the same hostname. Successive requests for a given hostname can be configured to return IP addresses in a round-robin fashion. Unified CVP's SIP design can make use of this capability in order to implement a
form of load balancing among multiple like components. Keep in mind that a DNS SRV lookup is performed for each and every call. If the DNS server is slow to respond, is unavailable, is across the WAN, or so forth, this will affect performance.

The DNS Server comes into play during SIP interactions in the following situations:

- When a call arrives at an Ingress Gateway, the dial peer can use DNS to alternate calls between the two SIP Proxy Servers. The SIP Proxy Servers can also use DNS to distribute incoming calls among multiple SIP Services. If SIP Proxy Servers are not being used, then the Ingress Gateway can use DNS directly to distribute inbound calls among multiple SIP Services.

- When the SIP Service is instructed by Unified ICME to transfer the call to the VRU leg, it can use DNS to alternate such requests between two SIP Proxy Servers. If SIP Proxy Servers are not being used, the SIP Service can use DNS directly to distribute VRU legs among multiple VoiceXML Gateways.

- When transferring a call to an agent using a SIP Proxy Server, the Cisco Unified Presence Server SIP Proxy cannot use DNS SRV for outbound calls; it must be configured with multiple static routes in order to do load balancing and failover. The static routes can point to an IP address or a regular DNS A host record. If SIP Proxy Servers are not being used, then the SIP Service can use DNS to locate the target agent's IP address.

Note that use of the DNS Server for SIP routing is entirely optional in Unified CVP. It is not required to have a dedicated DNS Server, but the existing DNS server needs to handle the additional load of Unified CVP. For every call destined for Unified CVP that comes into the network, there will be approximately 3 to 4 DNS lookups. You can determine the number of DNS queries per second by determining the number of calls per second for the solution, and multiplying that number by 4.

**Radvision IVP Server**

The Radvision Interactive Video Platform (IVP) is a media server platform for the development and delivery of enhanced interactive video communication services and applications. An IVP Server (IVP Controller) is a component of the Radvision IVP platform that communicates and controls the various components of the IVP, including the MCU and EMP. The IVP Server also communicates with the Unified CVP Call Server via an XML interface in order to notify Unified CVP of new calls and to receive instructions from Unified CVP on how to handle them, such as playing Video IVR prompts, transferring calls to agents, or streaming video files to callers as directed by an agent.

This component is used for Full Video Service deployments only.

**Content Services Switch**

The Content Services Switch (CSS) is the focus for high-availability design in the TCP arena. The CSS can be logically placed between one or more VoiceXML Gateways and one or more VoiceXML Servers, Media Servers, and ASR/TTS Servers. Various mechanisms allow the CSS to implement transparent load balancing and failover across these servers.

CSS is an optional device, but it is highly recommended. Without it, the IVR Service implements a "poor man's failover" mechanism, but it is not load-balanced and various retries and delays are part of the algorithm, all of which can be avoided if CSS is used.

The CSS is normally deployed as a Virtual Router Redundancy Protocol (VRRP) pair. It is useful in all deployment models except for Call Director call flows, which do not require use of VoiceXML Servers, Media Servers, or ASR/TTS servers. If SSL is used in the solution, you will need an SSL module for the CSS 11503 or 11506 chassis. The CSS 11501 has SSL support built in.
Third-Party Media Server

The media server component is a simple web server, such as Microsoft IIS or Apache, and is an optional component that can provide prerecorded audio files, external VoiceXML documents, or external ASR grammars to the gateway. Because some of these files can be stored in local flash memory on the gateways, the media server can be an optional component. However, in practice, most installations use a centralized media server to simplify distribution of prerecorded customer prompt updates. Media server functionality can also include a caching engine. The gateways themselves, however, can also do prompt caching when configured for caching. Typical media servers used are Microsoft IIS and Apache, both of which are HTTP-based.

As with ASR/TTS Servers, Media Servers may be deployed simplex, as a redundant pair, or with CSS in a farm. Note that the VoiceXML Gateway caches .wav files it retrieves from the Media Server. In most deployments, the Media Server encounters extremely low traffic from Unified CVP.

The Media Server can be installed co-resident with the Unified CVP Call Server or Unified CVP VoiceXML Server.


Third-Party Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) Servers

This component provides speech recognition services and text-to-speech services for the VoiceXML Gateway. Communication between the ASR/TTS server(s) and the VoiceXML Gateway uses Media Resource Control Protocol (MRCP). MRCP v1 can be used with both Micro-Apps and the VXML Server. MRCPv2 can be used only with the VXML Server.

For capacity and redundancy reasons, a CSS is usually used to mediate between a VoiceXML Gateway and a farm of ASR/TTS servers. If CSS is not used, then a VoiceXML Gateway can support a maximum of two ASR/TTS Servers.

Cisco does not sell, OEM, or support any ASR/TTS software or servers. Cisco does, however, test Unified CVP with ScanSoft, Nuance, and IBM offerings. A certification process is currently being developed to allow additional vendors to qualify their products against Unified CVP VoiceXML, and the World Wide Web Consortium (W3C) provides a rich feature set to support the ASR grammars. The simplest to implement and support is inline grammars, by which the set of acceptable customer responses is passed to the gateway. Another form is external grammars, by which Unified ICM passes a pointer to an external grammar source. The Unified CVP VoiceXML Server adds this pointer to the VoiceXML document that it sends to the VoiceXML Gateway, which then loads the grammar and uses it to check ASR input from the caller. In this case, the customer is responsible for creating the grammar file. A third type of grammar is the built-in grammar. For a complete explanation of grammar formats, consult the W3C website at http://www.w3.org/TR/speech-grammar/

The text for TTS is passed directly from the Unified CVP VoiceXML Server to the gateway. This action is referred to as inline TTS in this document.
The actual speech recognition and speech synthesis are performed by a separate server that interfaces directly to the VoiceXML gateway via Media Resource Control Protocol (MRCP). Currently, ScanSoft, Nuance, and IBM are the supported ASR/TTS engines. These ASR/TTS engines also support (with limitations) voice recognition and synthesis for multiple languages. For the latest information on supported languages and limitations of these ASR/TTS engines, refer to the following websites:

- Nuance and ScanSoft
  [http://www.nuance.com](http://www.nuance.com)
- IBM

Note that these are third-party products, which the customer or partner must purchase directly from the vendor. The customer also receives technical support directly from the vendor. That does not, however, mean that the vendor's latest software version can be used. Unified CVP is carefully tested with specific versions of each vendor's product, and Cisco Technical Assistance Center (TAC) will not support Unified CVP customers who use different ASR/TTS versions than those which have been tested with Cisco Unified CVP. For additional details on supported ASR and TTS products, consult the latest version of the [Hardware and System Software Specification for Cisco Unified CVP](http://www.cisco.com/en/US/products/sw/custcosw/ps1006/prod_technical_reference_list.html) (formerly called the Bill of Materials), available at


### Call Flows

This section describes the Unified CVP call flows for both SIP and H.323 calls.

#### Typical SIP Unified CVP Call Flow

A typical SIP Unified CVP call will arrive at an ingress voice gateway and send an invite message to the SIP Proxy Server, which will determine the IP address of the Unified CVP Server for the dialed number. The SIP Proxy Server will then forward the invite to the selected Unified CVP Server SIP Service. The SIP Service consults Unified ICME via the Unified CVP Server ICM Service. This consultation causes Unified ICME to run a routing script. The routing script will typically initiate a transfer of the call to a VoiceXML Gateway port via the SIP Service. The VoiceXML Gateway then sends a message to the IVR Service, which then requests scripted instructions from Unified ICME. Unified ICME then exchanges VRU instructions with the VoiceXML Gateway via the IVR Service. Among these VRU instructions can be requests to invoke more sophisticated applications on the VoiceXML Server. Such requests will result in multiple exchanges between the VoiceXML Server and the VoiceXML Gateway in order to provide self-service.

If the customer wants to transfer to a live agent, the Unified ICME routing script queues for an available agent. While waiting for an available agent, the Unified ICME provides additional instruction to the VoiceXML Gateway to provide queueing treatment to the caller. When an agent becomes available, Unified ICME sends a message to the Unified CVP Server SIP Service, which forwards a message via the SIP Proxy Server to the Ingress Gateway and to Unified CM to transfer the call away from the VoiceXML Gateway port and deliver it to the Unified CM agent IP phone.
Typical H.323 Unified CVP Call Flow

A typical H.323 Unified CVP call will arrive at an ingress voice gateway and send a route request to the H.323 Gatekeeper to determine to which Unified CVP Server to send this new call. The ingress gateway then interacts with the Unified CVP Server H.323 Service, which then consults Unified ICME via the Unified CVP Server IVR Service and Unified CVP Server ICM Service. This consultation causes Unified ICME to run a routing script. The routing script will typically initiate a transfer of the call to a VoiceXML Gateway port via the ICM, IVR, and H.323 Services. The VoiceXML Gateway then sends a message to the H.323 Service, which then requests scripted instructions from Unified ICME via the IVR and ICM Services. Unified ICME then exchanges VRU instructions with the VoiceXML Gateway via the ICM, IVR, and H.323 IVR Services. Among these VRU instructions can be requests to invoke more sophisticated applications on the VoiceXML Server. Such requests will result in multiple exchanges between the VoiceXML Server and the VoiceXML Gateway in order to provide self-service.

If the customer wants to transfer to a live agent, the Unified ICME routing script queues for an available agent. While waiting for an available agent, the Unified ICME provides additional instruction to the VoiceXML Gateway to provide queueing treatment to the caller. When an agent becomes available, Unified ICME sends a message to the Unified CVP Server H.323 Service, which then queries the H.323 gatekeeper for routing instructions to the appropriate Unified CM H.323 trunk. Then the H.323 Service signals to the Ingress Gateway and Unified CM to transfer the call away from the VoiceXML Gateway port and deliver it to the Unified CM agent IP phone.

Design Process

Unified CVP has an infinite number of possible deployment models due to it’s modularity, flexibility, resiliency, and scalability.

When designing a Unified CVP deployment, it is most important first to choose a functional deployment model. After choosing a functional deployment model, Unified CVP solution designers must determine where the Unified CVP components are going to be deployed (in the data center or at a branch). Then Unified CVP solution designers much choose the amount of availability and resiliency that is justifiable or required. Finally, Unified CVP solution designers must size the deployment to provide the justifiable or required grade of service for the initial deployment and near-term growth.

Choosing SIP or H.323

When choosing the call control protocol for your Unified CVP implementation, the following factors can have a direct influence on your decision:

- Characteristics of using SIP with Unified CVP
  - SIP is the preferred call control protocol for Unified CVP. New features are not planned for H.323.
  - SIP supports a higher port capacity per Unified CVP call server.
  - SIP can use a static dial plan on the Call Server (suitable for simple deployments).
  - Transferred legs use a Ring-No-Answer timeout based on the destination pattern (rather than the single server-wide setting used with H.323)
  - Third-party SIP endpoints are supported (subject to testing).
  - Video calls are supported.
- Failover and load balancing are performed with the SRV mechanism on a per-call basis, but no heartbeats are maintained and endpoints are not registered.

- Characteristics of using H.323 with Unified CVP:
  - H.323 provides stability and product maturity.
  - H.323 supports post-call survey.
  - H.323 provides load balancing and failover using a gatekeeper, with registered endpoints and a heartbeat mechanism (thus providing faster failover time).
  - GKTMP is supported.
  - Hookflash transfers are supported.

### Functional Deployment Models

As mentioned previously, the first step in the design process is typically to determine what functionality you need. Unified CVP offers the following functional deployment models:

- Standalone VoiceXML Server — Provides a standalone VRU with no integration to Unified ICM for queuing control or subsequent call control.
- Call Director — Provides IP switching services only.
- Comprehensive — Provides IVR services, queue treatment, and IP switching services. The previously described typical call flows use this functional deployment model.
- VRU Only — Provides IVR services, queueing treatment, and switching for SS7/IN PSTN endpoints. This model relies upon the PSTN to transfer calls between call termination endpoints.

For more details and design considerations for each of these functional deployment models, see the chapter on Functional Deployment Models, page 2-1.

### Distributed Network Options

After choosing a functional deployment model, Unified CVP solution designers must determine where the Unified CVP components will be deployed. Unified CVP deployment can use one of the following primary distributed network options:

- Combined Branch Gateways — Allows for call treatment at the edge and integration of locally dialed numbers into the enterprise virtual contact center. This option can be either a combined Ingress and VoiceXML gateway or separate gateways, but typically they are combined when deployed in a branch.
- Branch Ingress Gateways with Centralized VoiceXML Gateways — Allows for integration of locally dialed numbers and resource grouping of VoiceXML gateways. This option might be desirable for organizations with many medium to large branches but with few contact center calls in those branches. However, VRU announcements would have to traverse the WAN from the VoiceXML Gateway to the Ingress Gateway.
- Branch Egress Gateways — Allows for calls to be transferred across the WAN to remote TDM terminations.
- Branch Agents — Allows for a virtual contact center where agents can be located anywhere on the IP network.
It is also possible to use a combination of these distributed options. For more details and design considerations for each of these distributed network options, see the chapter on Distributed Deployments, page 3-1.

High Availability Options

After choosing a functional deployment model and distributed deployment options, Unified CVP solution designers must choose the amount of availability required. Unified CVP solution designers can increase solution availability in the following areas:

- Multiple gateways, Unified CVP Servers, VoiceXML Servers, VRU PGs, Cisco Unified Presence Servers, and gatekeepers — Allow for inbound and outbound call processing and IVR services to continue upon component failure.
- Multiple call processing locations — Allow for call processing to continue in the event of a loss of another call processing location.
- Redundant WAN links — Allow Unified CVP call processing to occur upon failure of individual WAN links.
- Content Services Switches — Provide an efficient means to remove failed Unified CVP Servers, VoiceXML Servers, and Media Servers from the load-balancing algorithms being used for those components.

It is also possible to use a combination of these high availability options to be utilized. For more details and design considerations for each of these high-availability network options, see the chapter on Designing Unified CVP for High Availability, page 4-1.

Scalability Options

After choosing the functional model and the distributed and high-availability deployment options, Unified CVP solution designers must then size their solution and select appropriate hardware. To make Unified CVP deployments larger, Unified CVP supports multiple gateways, Unified CVP Servers, and VoiceXML Servers. To load-balance HTTP requests efficiently to multiple Unified CVP Servers, VoiceXML Servers, and media stores, you can use the Content Services Switch (CSS). For more details on choosing appropriate hardware for your deployment, see the chapter on Sizing, page 14-1.
Chapter 2

Functional Deployment Models

This chapter covers the following functional deployment models for Unified CVP:

- Standalone VoiceXML Server, page 2-1
- Call Director, page 2-3
- Comprehensive, page 2-6
- VRU Only, page 2-10

For each model, this chapter provides a short discussion of the typical customer requirements for that deployment model, a list of the required and optional components, and a step-by-step call flow.

The functional deployment models presented in this chapter assume all components are located in a single site, and no discussion of failover is covered. Distributed deployment scenarios where components are separated across a WAN link are discussed in the chapter on Distributed Deployments, page 3-1. High-availability deployment options are covered in the chapter on Designing Unified CVP for High Availability, page 4-1.

Standalone VoiceXML Server

This deployment model is the simplest of the Unified CVP functional deployment models. It provides organizations with a standalone IVR solution for automated self-service. Callers can access Unified CVP via either local, long distance, or toll-free numbers terminating at Unified CVP Ingress voice gateways. Callers can also access Unified CVP from VoIP endpoints. Figure 2-1 illustrates this model.

Figure 2-1 Functional Deployment Model for a Standalone VoiceXML Server
This model requires the following components:

- Ingress voice gateway(s)
- VoiceXML gateway(s) (Can be co-resident with the ingress gateway)
- VoiceXML server(s)
- VoiceXML Studio
- Operations Console Server

Optional components for this model include:

- ASR/TTS server(s)
- Third-party media server(s)
- Content Services Switch(es)
- Egress voice gateway(s)
- Unified CVP Reporting Server

**Protocol-Level Call Flow**

1. A call arrives at the ingress gateway via TDM, SIP, or H.323. The gateway performs normal inbound POTS or VoIP dial-peer matching.

2. The selected VoiceXML gateway port invokes the Unified CVP self-service TCL script.

3. The TCL script invokes the Unified CVP standalone bootstrap VoiceXML Document, which performs an HTTP request to the configured IP address of the VoiceXML server.

4. The VoiceXML Server runs the application specified in the HTTP URL and returns a dynamically generated VoiceXML document to the VoiceXML gateway. The VoiceXML Server application may access back-end systems to incorporate personalized data into the VoiceXML document that is sent to the VoiceXML gateway.

5. The VoiceXML gateway parses and renders the VoiceXML document. For spoken output, the VoiceXML gateway either retrieves and plays back prerecorded audio files referenced in the VoiceXML document, or it streams media from a text-to-speech (TTS) server. Caller input can be captured either by DTMF detection on the Ingress Gateway or via DTMF/speech recognition on an ASR server.

6. As defined in the VoiceXML document, the VoiceXML gateway submits an HTTP request containing the results of the caller input to the VoiceXML Server. The VoiceXML Server again runs the application specified in the HTTP URL and returns a dynamically generated VoiceXML document to the VoiceXML gateway. The dialog continues by repeating steps 5 and 6.

7. The IVR dialogue ends when either the caller hangs up, the application releases, or the application initiates a transfer.

**Transfers and Subsequent Call Control**

In addition to providing self-service, the Standalone VoiceXML deployment model can transfer callers to another endpoint – either VoIP (for example, Cisco Unified Communications Manager) or TDM (for example, egress voice gateway to PSTN or TDM ACD). However, no IVR application data can be passed to the new endpoint with this deployment model, therefore there will be no agent screen pop if the endpoint is a TDM ACD.
This model supports the following types of transfers:

- VoiceXML Bridged Transfer
- VoiceXML Blind Transfer
- Release Trunk Transfer (TNT, hookflash, TBCT, and SIP Refer)

The VoiceXML transfers are invoked using the VoiceXML Studio’s `transfer` element. Release Trunk Transfers are invoked by providing specially formatted return values in VoiceXML Studio’s `subdialog_return` element.

In the case of a VoiceXML Bridged Transfer, the outcome of the transferred call leg (transfer failed, transfer call leg released, and so forth) is submitted back to the VoiceXML server. The VoiceXML session is then resumed, and further iterations of IVR call treatment and transfers can be performed. During the period of time that the call is transferred, a VoiceXML server port license is utilized with a bridged transfer.

In the case of a VoiceXML 2.0 Blind Transfer, the call remains connected through the ingress voice gateway, but Unified CVP does not have any method to provide any subsequent call control.

In the case of a Release Trunk Transfer, the ingress voice gateway port is released and no subsequent call control is possible.

For more details on transfers, see to the chapter on Call Transfer Options, page 10-1.

Call Director

This functional deployment model provides organizations with a mechanism to route and transfer calls across a VoIP network. The most common usage scenario for this model is for organizations with multiple TDM ACD and TDM IVR locations that are integrated with Unified ICM via an ACD or IVR PG, and they wish to use Unified ICM to route and transfer calls intelligently across these locations without having to utilize PSTN pre-routing or release trunk transfer services. In this functional deployment model, Unified CVP and Unified ICM can also pass call data between these ACD and IVR locations. In this deployment model, Unified ICM can also provide cradle-to-grave reporting for all calls.

This functional deployment model is often the initial step in the migration from a TDM-based contact center to a VoIP-based contact center. When the organization is ready to implement CVP-based IVR services and Cisco Unified Contact Center Enterprise, the organization can migrate their Unified CVP deployment to the comprehensive functional deployment model.

Callers can access Unified CVP via either local, long distance, or toll-free numbers terminating at Unified CVP ingress voice gateways. Callers can also access Unified CVP from VoIP endpoints.

Call Director deployments can utilize either H.323, SIP, or a combination.

This model requires the following components:

- Ingress voice gateway(s)
- Egress voice gateway(s)
- Unified CVP Server
- Unified CVP Operations Console Server
- Cisco Unified ICM Enterprise
- H.323 gatekeeper (H.323 deployments)
- SIP Proxy Server (for SIP deployments)
Optional components for this model include:

- Unified CVP Reporting Server

## SIP Protocol-Level Call Flow

### VoIP-based Pre-Routing

1. A call arrives at the ingress gateway and sends a SIP INVITE message to the SIP Proxy Server, which forwards the request to the Unified CVP Server SIP Service.

2. The SIP Service sends a route request to Unified ICM via the Unified CVP Server ICM Service and the VRU PG. This route request causes Cisco Unified ICM to run a routing script based upon the dialed number and other criteria.

3. The Unified ICM routing script selects a target and returns a translation route label to the Unified CVP Server SIP Service, which then signals via the SIP Proxy Server to the egress voice gateway (which connects to the TDM termination) and the ingress voice gateway to enable the call to be set up between the ingress and egress voice gateways. While the RTP stream flows directly between the ingress and egress voice gateways, call control signaling flows through Unified CVP in order to allow subsequent call control.

4. When the call arrives at the selected termination, the termination equipment sends a request to its PG for routing instructions. This step resolves the translation route and allows any call data from the previously run Unified ICM script to be passed to the selected termination. If the selected termination is a TDM IVR platform, then self-service will be provided and the caller can either release or request to be transferred to a live agent. If the selected termination is a TDM ACD platform, then the caller will be queued until an available agent is selected by the TDM ACD. Call data can then be popped onto the agent screen. After receiving live assistance, the caller can either release or request to be transferred to another agent.

### VoIP-based Transfer

1. Regardless of whether the call was initially routed to a TDM IVR or ACD location, the caller can request to be transferred to another location. When this occurs, the TDM IVR or ACD sends a post-route request with call data (via its PG) to Cisco Unified ICM.

2. When Unified ICM receives this post-route request, it runs a routing script based upon the transfer dialed number and other criteria. The Unified ICM routing script selects a new target for the call and then signals to the Unified CVP Server SIP Service to release the call leg to the originally selected termination and to extend the call to a new termination.

3. When the call arrives at the new termination, the termination equipment sends a request to its PG for routing instructions. This step resolves a translation route that was allocated for this call to this new termination location, and it allows any call data from the previous location (IVR port or agent) to be passed to the new termination. Calls can continue to be transferred between locations using this same VoIP-based transfer call flow.
H.323 Protocol-Level Call Flow

VolP-based Pre-Routing
1. A call arrives at the ingress gateway and sends a RAS request to the H.323 gatekeeper to find the IP address of an appropriate Unified CVP Server for that dialed number.
2. The ingress voice gateway then sends an H.225 call setup message to the Unified CVP Server H.323 Service. For a brief instance, a G.711 voice stream exists between the ingress voice gateway and the Unified CVP Server H.323 Service.
3. The Unified CVP Server H.323 Service sends a route request to Cisco Unified ICM via the Unified CVP Server IVR Service, Unified CVP ICM Service, and VRU PG. This request causes Unified ICM to run a routing script based upon the dialed number and other criteria.
4. The Unified ICM routing script selects a target and returns a translation route label (dialed number) to the Unified CVP Server H.323 Service, which then sends a RAS request to the H.323 gatekeeper to find the IP address of the selected termination (an egress voice gateway to the PSTN or front-ending a TDM peripheral).
5. The Unified CVP Server H.323 Service then sends an H.225 call setup message to the egress voice gateway and makes an Empty Capability Set (ECS) request to the ingress voice gateway to redirect the call. While the RTP stream flows directly between the ingress and egress voice gateways, call control signaling flows through Unified CVP in order to allow subsequent call control.
6. When the call arrives at the selected termination, the termination equipment sends a request to its PG for routing instructions. This step resolves the translation route and allows any call data from the previously run Unified ICM script to be passed to the selected termination. If the selected termination is a TDM IVR platform, then self-service will be provided and the caller can either release or request to be transferred to a live agent. If the selected termination is a TDM ACD platform, then the caller will be queued until an available agent is selected by the TDM ACD. Call data can then be popped onto the agent screen. After receiving live assistance, the caller can either release or request to be transferred to another agent.

VolP-based Transfer
1. Regardless of whether the call was initially routed to a TDM IVR or ACD location, the caller can request to be transferred to another location. When this occurs, the TDM IVR or ACD will send a post-route request with call data (via its PG) to Cisco Unified ICM.
2. When Unified ICM receives this post-route request, it runs a routing script based upon the transfer dialed number and other criteria. The Unified ICM routing script selects a new target for the call and then signals to the Unified CVP Server H.323 Service to release the call leg to the originally selected termination and to extend the call to a new termination. The H.323 Service queries the H.323 gatekeeper in order to get an IP address for the new termination.
3. When the call arrives at the new termination, the termination equipment sends a request to its PG for routing instructions. This step resolves a translation route that was allocated for this call to this new termination location, and it allows any call data from the previous location (IVR port or agent) to be passed to the new termination. Calls can continue to be transferred between locations using this same VoIP-based transfer call flow.

Transfers and Subsequent Call Control

In addition to the transfers managed by Unified ICM (as described above), the Call Director deployment model can transfer calls to non-ICM terminations or invoke a Release Trunk Transfer in the PSTN. If a call is transferred to a non-ICM termination, then no call data can be passed to the termination, no further
call control is possible for that call, and the cradle-to-grave call reporting that Unified ICM captures is completed. In the case of a Release Trunk Transfer, the ingress voice gateway port is released, no call data can be passed to the termination, and no further call control is possible for that call. If the Release Trunk Transfer was translation-routed to another ICM peripheral, call data and cradle-to-grave reporting can be maintained. For more details on transfers, see the chapter on Call Transfer Options, page 10-1.

If a selected termination (for either a new or transferred call) returns a connection failure or busy status, or if the target rings for a period of time that exceeds the Unified CVP Call Server's ring-no-answer (RNA) timeout setting, the Unified CVP Call Server cancels the transfer request and sends a transfer failure indication to Unified ICM. This scenario causes a Router Requery operation. The Unified ICM routing script then recovers control and has the opportunity to select a different target or take other remedial action.

**Comprehensive**

This functional deployment model provides organizations with a mechanism to route and transfer calls across a VoIP network, to offer IVR services, and to queue calls before being routed to a selected agent. The most common usage scenario for this functional deployment model is for organizations wanting a pure IP-based contact center. Callers are provided IVR services initially and then, upon request, are provided queue treatment and are transferred to a selected Unified CCE agent. Upon request, callers can also be transferred between Unified CCE agents. In this functional deployment model, Unified CVP and Unified ICM can also pass call data between these endpoints and provide cradle-to-grave reporting for all calls. Figure 2-2 illustrates this model.

*Figure 2-2  Comprehensive Functional Deployment Model*

This functional deployment model provides all the capabilities of the Standalone VoiceXML Server and Call Director functional deployment models, plus the ability to route and queue calls to Unified CCE agents.

Callers can access Unified CVP via either local, long distance, or toll-free numbers terminating at the Unified CVP ingress voice gateways. Callers can also access Unified CVP from VoIP endpoints. Comprehensive deployments can utilize either H.323, SIP, or a combination.
This model requires the following components:

- Ingress voice gateway(s)
- VoiceXML gateway(s) (Can be co-resident with the ingress gateway)
- Unified CVP Server
- Unified CVP Operations Console Server
- Cisco Unified ICM Enterprise
- H.323 gatekeeper (for H.323 deployments)
- SIP Proxy Server (for SIP deployments)

Optional components for this model include:

- VoiceXML Server
- Egress voice gateway(s)
- ASR / TTS server(s)
- Third-party media server(s)
- Content Services Switch(es)
- Unified CVP Reporting Server

**SIP Protocol-Level Call Flow**

**Initial Call Treatment and Self-Service**

1. A call arrives at the ingress gateway and sends a SIP invite message to the SIP Proxy Server, which forwards the request to the Unified CVP Server SIP Service.

2. The SIP Service sends a route request to Unified ICM via the Unified CVP Server ICM Service and the VRU PG. This route request causes Cisco Unified ICM to run a routing script based upon the dialed number and other criteria.

3. The Unified ICM routing script utilizes a Send to VRU node to return a label to the SIP Service to have the call sent to a VoiceXML gateway. The Unified CVP Server SIP Service sends an invite message to the VoiceXML gateway via the SIP Proxy Server, which translates the label DN to the IP address of the VoiceXML gateway.

4. The Voice XML gateway sends an HTTP new-call message to the Unified CVP Server IVR Service with the label DN provided by Unified ICM. The IVR Service then sends a route request message to Unified ICM (via the Unified ICM Service), which then allows Unified ICM to re-enter the previously started routing script. The routing script is re-entered at the successful exit path of the Send to VRU node. The Unified ICM routing script then uses Run Script nodes to instruct the IVR service about the desired call treatment. If call treatment requires complex IVR self-services, service control can be redirected to a VoiceXML server application. Upon completion of the VoiceXML Server application or a request by the caller to transfer to a live agent, service control is returned to the Unified CVP Server IVR Service. If the initial call treatment is simple with just a few prompts, then the IVR Service can utilize Unified CVP microapplications to generate VoiceXML documents for the VoiceXML gateway, and a VoiceXML Server is not required.

**Caller Requests to Transfer to Live Agent**

1. When the caller requests to transfer to a live agent, the Unified ICM routing script queues the caller for an appropriate skill group and sends Run VRU Script messages to the IVR Service to have queue treatment provided (assuming no agent is available).
2. When a Unified CCE agent becomes available, Unified ICM requests the Unified CVP Server IVR Service to transfer the call to the selected agent.

3. The IVR Service then requests the SIP Service to transfer the caller to the dialed number of the selected agent. The SIP Service then sends a SIP invite message to the SIP Proxy Server, which finds the Cisco Unified Communications Manager SIP Trunk IP address associated with this agent DN, and then forwards the SIP Invite message to Cisco Unified Communications Manager (Unified CM).

4. Unified CM accepts the incoming SIP Trunk call and routes it to the selected agent.

**Caller Requests to be Transferred to a Second Skill Group**

1. If the caller requests to be transferred to a second agent, then the first agent will initiate a transfer from their Unified CCE agent desktop application. This action generates a route request from the agent PG to the Unified ICM central controller. Unified ICM then executes a routing script that queues the call to another skill group. Assuming no agent is available, the Unified ICM script will use the Send to VRU node, which will signal to the SIP Service to release the call leg to the Unified CM SIP Trunk and connect the call back to a VoiceXML gateway.

2. The VoiceXML gateway sends an HTTP new-call request to the IVR Service, which forwards that request to Unified ICM in order to allow the routing script to be re-entered at the exit of the Send to VRU node. Unified ICM then sends Run VRU Script messages to the IVR Service to allow queue treatment to be provided to the caller while waiting for a second agent.

3. When a second Unified CCE agent becomes available, Unified ICM requests the Unified CVP Server IVR Service to transfer the call to the selected agent.

4. The IVR Service then requests the SIP Service to transfer the caller to the dialed number of the selected agent. The SIP Service then sends a SIP invite message to the SIP Proxy Server, which finds the Unified CM SIP Trunk IP address associated with the second agent DN, and then forwards the SIP Invite message to Unified CM.

5. Unified CM accepts the incoming SIP trunk call and routes it to the second agent.

---

**Note**

Due to a limitation in Cisco IOS, Cisco recommends configuring an MTP because it will be required for call flows in which the first agent consults and gets queued and then completes the transfer before connecting to a second agent. The MTP usage can be minimized by creating an additional SIP trunk with an MTP configured on it from Unified CM to Unified CVP or proxy server calls. This will allow all calls going from Unified CVP to Unified CM to use the first trunk without MTP allocation, and only the transfers with queuing (not Unified ICM network transfers) will involve the trunk with MTP allocation.

---

**H.323 Protocol-Level Call Flow**

**Initial Call Treatment and Self-Service**

1. A call arrives at the ingress gateway and sends a RAS request to the H.323 gatekeeper to find the IP address of the Unified CVP Server for that dialed number.

2. The ingress voice gateway then sends an H.225 call setup message to the Unified CVP Server H.323 Service. For a brief instance, a G.711 voice stream exists between the ingress voice gateway and the Unified CVP Server H.323 Service.

3. The Unified CVP Server H.323 Service sends a route request to Cisco Unified ICM via the Unified CVP Server IVR Service, Unified CVP ICM Service, and VRU PG. This request causes Unified ICM to run a routing script based upon the dialed number and other criteria.
4. The Unified ICM routing script utilizes a Send to VRU node to return a label to the H.323 Service to have the call sent to a VoiceXML gateway. The H.323 Service sends a RAS request to the H.323 gatekeeper to find the IP Address of the VoiceXML gateway associated with the label returned by Unified ICM. The Unified CVP H.323 Service sends an H.225 setup message to the VoiceXML Gateway.

5. The Voice XML gateway sends an HTTP new-call message to the Unified CVP Server IVR Service, with the label DN provided by Unified ICM. The IVR Service then sends a route request message to Unified ICM (via the IVR Service, ICM Service, and VRU PG), which then allows Unified ICM to re-enter the previously started routing script. The routing script is re-entered at the successful exit path of the Send to VRU node. The Unified ICM routing script then uses Run Script nodes to instruct the IVR service about the desired call treatment. If call treatment requires complex IVR self-services, service control can be redirected to a VoiceXML server application. Upon completion of the VoiceXML Server application or a request by the caller to transfer to a live agent, service control is returned to the Unified CVP Server IVR Service. If the initial call treatment is simple with just a few prompts, then the IVR Service can utilize Unified CVP microapplications to generate VoiceXML documents for the VoiceXML gateway, and a VoiceXML Server is not required.

**Caller Requests to Transfer to Live Agent**

1. When the caller requests to transfer to a live agent, the Unified ICM routing script queues the caller for an appropriate skill group and sends Run VRU Script messages to the IVR Service to have queue treatment provided (assuming no agent is available).

2. When a Unified CCE agent becomes available, Unified ICM requests the Unified CVP Server IVR Service to transfer the call the selected agent.

3. The IVR Service then requests the H.323 Service to transfer the caller to the dialed number of the selected agent. The H.323 Service then sends a RAS message to the H.323 gatekeeper to find the Unified CM H.323 Trunk IP address associated with this agent DN, and then sends an H.225 call setup message to Unified CM.

4. Unified CM accepts the incoming H.323 trunk call and routes it to the selected agent.

**Caller Requests to be Transferred to a Second Skill Group**

1. If the caller requests to be transferred to a second agent, then the first agent will initiate a transfer from their Unified CCE agent desktop application. This action generates a route request from the agent PG to the Unified ICM central controller. Unified ICM then executes a routing script that queues the call to another skill group. Assuming no agent is available, the Unified ICM script will use the Send to VRU node, which signals to the H.323 Service to release the call leg to the Unified CM H.323 Trunk and to connect the call back to a VoiceXML gateway. A RAS request to the H.323 gatekeeper is to find the IP address of the VoiceXML gateway.

2. The VoiceXML gateway sends an HTTP new-call request to the IVR Service, which forwards that request to Unified ICM in order to allow the routing script to be re-entered at the exit of the Send to VRU node. Unified ICM then sends Run VRU Script messages to the IVR Service to allow queue treatment to be provided to the caller while waiting for a second agent.

3. When a second Unified CCE agent becomes available, Unified ICM requests the Unified CVP Server IVR Service to transfer the call the selected agent.

4. The IVR Service then requests the H.323 Service to transfer the caller to the dialed number of the selected agent. The H.323 Service sends a RAS request to the H.323 gatekeeper to get the IP address of the Unified CM H.323 trunk associated with the second agent DN. The H.323 Service then sends an H.225 setup message to Unified CM.

5. Unified CM accepts the incoming H.323 trunk call and routes it to the second agent.
**Chapter 2  Functional Deployment Models**

**Transfers and Subsequent Call Control**

In addition to transfers manager by Unified ICM (as described above), the Comprehensive deployment model can transfer calls to non-ICM terminations or it can invoke a Release Trunk Transfer in the PSTN. If a call is transferred to a non-ICM termination, then no call data can be passed to the termination, no further call control is possible for that call, and the cradle-to-grave call reporting that Unified ICM captures is completed. In the case of a Release Trunk Transfer, the ingress voice gateway port is released, no call data can be passed to the termination, and no further call control is possible for that call. If the Release Trunk Transfer was translation-routed to another ICM peripheral, call data and cradle-to-grave reporting can be maintained. For more details on transfers, see the chapter on Call Transfer Options, page 10-1.

If a selected termination (for either a new or transferred call) returns a connection failure or busy status, or if the target rings for a period of time that exceeds the Unified CVP Call Server's ring-no-answer (RNA) timeout setting, the Unified CVP Call Server cancels the transfer request and sends a transfer failure indication to Unified ICM. This scenario causes a Router Requery operation. The Unified ICM routing script then recovers control and has the opportunity to select a different target or take other remedial action.

**VRU Only**

This functional deployment model provides self-service applications and queueing treatment for organizations that are utilizing advanced PSTN switching services that are controlled via a Cisco Unified ICM PSTN Network Interface Controller (NIC). Two Unified ICM PSTN NICs are available that allow subsequent call control of calls in the PSTN. They are the SS7 NIC and the Carrier Routing Service Protocol (CRSP) NIC. These NICs go beyond allowing Unified ICM to pre-route calls intelligently to Unified ICM peripherals (such as ACDs and IVRs); they also allow Unified ICM to invoke mid-call transfers in the PSTN. Figure 2-3 illustrates this model.
A typical call in this model would be pre-routed by Unified ICM to a Unified CVP Ingress Voice Gateway for call treatment and queuing. When an agent becomes available, Unified ICM instructs the PSTN to transfer the call to that agent. The agents can be Cisco Unified Contact Center Enterprise agents, Cisco Unified Contact Center Express agents, or traditional ACD agents. If necessary, Unified ICM can request the PSTN (via the NIC) to transfer the call again and again, just as Unified ICM can request Unified CVP to transfer the call again and again. In this functional deployment model, the Unified CVP Ingress Voice Gateway is just a Unified ICM-managed PSTN termination point that is capable of providing VURI services via a VoiceXML gateway, the Unified CVP Server IVR Service, the Unified CVP Server ICM Service, and Unified ICM. In this functional deployment model, neither the Unified CVP Server H.323 Service nor the Unified CVP Server SIP Service is used for call control. All call control and switching is controlled by Unified ICM and the PSTN. In this functional deployment model, Unified ICM can pass call data between these termination points (for a screen pop or other intelligent treatment) and provide cradle-to-grave reporting for all calls.

This model requires the following components:

- Ingress voice gateway(s)
- VoiceXML gateway(s) (Can be co-resident with the ingress gateway)
- Unified CVP Server running IVR Service and ICM Service
- Unified CVP Operations Console Server
- Cisco Unified ICM Enterprise and NIC (SS7 or CRSP)

Optional components for this model include:

- VoiceXML Server
- ASR / TTS server(s)
- Third-party media server(s)
- Content Services Switch(es)
- Unified CVP Reporting Server
- H.323 gatekeeper (for H.323 deployments)
- SIP Proxy Server (for SIP deployments)

### Protocol-Level Call Flow

#### Initial Call Treatment and Self-Service

1. A call arrives at the PSTN, and the PSTN sends a new-call message to Unified ICM via either a CRSP NIC or SS7 NIC. Unified ICM invokes a routing script based upon the dialed number, and the routing script uses either a Send to VURI node or a Translation Route to VURI node to send a result to the PSTN to have the call routed to the Unified CVP ingress voice gateway. Depending upon the PSTN capability and Unified ICM VURI type for the Unified CVP deployment, the response returned to the PSTN is either a translation route label (dialed number) or a dialed number plus correlation ID.

2. The PSTN routes the call to an available ingress voice gateway port. The ingress voice gateway performs normal inbound POTS dial-peer matching to deliver the call to an available VoiceXML gateway port. An H.323 RAS request to an H.323 gatekeeper or a SIP Invite message to a SIP Proxy server could be used to aid in the routing of the call to an available VoiceXML gateway port, if desired.
3. The Voice XML gateway sends an HTTP new-call message to the Unified CVP Server IVR Service with the dialed number delivered from the PSTN. This dialed number represents either a translation route label or a correlation ID. Either way, the Unified ICM VRU PG will recognize this call and send a request instruction message to the in-progress Unified ICM routing script. The next routing script node is typically a Run VRU Script node to instruct the VRU which microapplication is to be executed.

4. The Unified CVP Server IVR Service sends a dynamically generated VoiceXML document to the VoiceXML gateway for rendering.

5. The VoiceXML gateway parses and renders the VoiceXML document. If call treatment requires complex IVR self-services, service control can be redirected to a VoiceXML server application. Upon completion of the VoiceXML Server application or a request by the caller to transfer to a live agent, service control is returned to Unified CVP Server IVR Service. If the initial call treatment is simple with just a few prompts, then the IVR Service can utilize Unified CVP microapplications to generate VoiceXML documents for the VoiceXML gateway, and a VoiceXML Server is not required. If desired, the Unified ICM routing script can terminate the call, and a disconnect message will be sent by the Unified ICM to the PSTN via the PSTN NIC.

**Caller Requests to Transfer to Live Agent**

6. When the caller requests to transfer to a live agent, the Unified ICM routing script queues the caller for an appropriate skill group and sends Run VRU Script messages to the IVR Service to have queue treatment provided (assuming no agent is available).

7. When a Unified CCE agent or a TDM ACD agent becomes available, Unified ICM immediately sends a connect message to the PSTN via the PSTN NIC. The connect message will contain either a translation route label or a dialed number plus correlation ID (depending upon the PSTN capabilities). Upon receipt of the connect request, the PSTN releases the call leg to the Unified CVP ingress voice gateway and connects the call to the new termination. If the new termination is a TDM ACD, the previous queuing treatment could be skipped and the TDM ACD could provide the queue treatment. Any call data associated with this call will be passed to the Unified ICM Peripheral Gateway (PG) for the selected peripheral.

**Caller Requests to be Transferred to a Second Skill Group**

8. If the caller requests to be transferred to a second agent, then the first agent will initiate a transfer from their agent desktop application (Unified CCE or TDM). This action generates a route request from the PG to the Unified ICM central controller.

9. Unified ICM executes a routing script. If the caller needs to be placed back into queue on Unified CVP or to another ACD location (TDM or IP), then Unified ICM sends a connect message to the PSTN via the PSTN NIC to have the call transferred. If the caller needs to be transferred to an agent on the same Unified CM peripheral, then Unified ICM instructs Unified CM (via the Unified CM PG) to transfer the call.
Basic Video Service

The Basic Video Service is simply an extension of the existing Comprehensive deployment model, but it allows for a video caller to interact with a video agent. IVR and queuing are audio-only.

The following video endpoints are supported when using the Unified CVP Basic Video Service:

- Cisco Unified IP Phone 7985G
- Cisco Unified Video Advantage
- Cisco TelePresence

The Basic Video Service supports the following call flows with Cisco TelePresence:

- A TelePresence caller dials into Unified CVP, receives audio-only IVR and/or queuing treatment, and then is transferred to an Agent on a second TelePresence unit.
- A TelePresence caller dials into Unified CVP, receives audio-only IVR and/or queuing treatment, and then is transferred to an Agent on a second TelePresence unit. The TelePresence Agent can conference in a second Agent on an audio-only IP phone by dialing a direct extension from their TelePresence phone.
- A TelePresence caller dials into Unified CVP, receives audio-only IVR and/or queuing treatment, and then is transferred to an Agent on a second TelePresence unit. The TelePresence Agent can conference in a Unified CVP dialed number that results in audio queuing, followed by connection to a second Agent on an audio-only IP phone.
- A TelePresence caller dials into Unified CVP, receives audio-only IVR and/or queuing treatment, and then is transferred to an Agent on an audio-only IP Phone. MTP must be enabled on the SIP trunk or else one-way audio is encountered.

Because the Basic Video Service is simply an extension of the SIP-based Comprehensive deployment model, the required components and SIP protocol-level call flow details are identical.

Full Video Service

The Full Video call flow model is for customers who want to use video capabilities for every phase of the caller interaction with the call center. Figure 2-4 illustrates this model.
Using the Radvision IVP platform in conjunction with Unified CVP and Cisco Unified Intelligent Contact Management Enterprise (Unified ICME) provides the following video capabilities:

- **Video IVR** — Callers can see a visual representation of their menu choices, they can perform self-service functions, and they can select and view pre-recorded video clips or live video sources.
- **Video Queuing** — Callers can view pre-recorded or live video while in queue for an agent.
- **Video Agent** — Once the caller is connected to an agent, the agent and caller can communicate using video. Additionally, agents have the ability to select and “push” pre-recorded or live video content to the caller.
- **Video Hold** — If the agent puts the caller on hold, the caller will view pre-recorded video clips and/or live streaming video and hear the Unified CM music on hold.
- **Video Recording** — Any phase of the video interactions outlined above can be recorded.

The following video endpoints are supported when using the Unified CVP Full Video Service:

- Cisco Unified IP Phone 7985G
- Cisco Unified Video Advantage
- Radvision 3G Video Gateway and 3G Mobile Video Phone

---

**Figure 2-4** Functional Deployment Model for Full Service Video
Protocol-Level Call Flow

The following step outline the call flow for Full Video Service.

Initial Call Treatment and Self-Service
1. A call arrives from a video-enabled endpoint. Either the 3G gateway (mobile phone caller) or Unified CM sends a SIP invite to the Radvision IVP via the Cisco Unified Presence proxy server.
2. The IVP sends an XML message to the Unified CVP call server notifying it of the new call.
3. Unified CVP sends a GED request to Unified ICM, which runs a script based on the DNIS of the new call. Unified ICM returns GED responses to Unified CVP, which translates the GED messages into IVP XML messages and sends them back to the IVP.
4. The IVP, through its own internal mechanisms, plays movies or collects DTMF tones, depending on the commands sent from Unified CVP. Responses from the action are sent back to Unified CVP, which in turn relays the responses to the Unified ICM script. If the script writer has enabled VCR controls on a microapplication definition, the caller may elect to stop, pause, resume, forward, or rewind the movie using administrator-configurable DTMF keys.
5. The Unified ICM script writer can set an ECC variable to start and stop recording the caller view, the agent view, or the omni view (meaning all call legs).

Caller Requests to Transfer to Live Agent
6. When an agent becomes available, Unified ICM sends a CONNECT request to Unified CVP, with instructions to transfer the caller to an agent extension. Unified CVP translates this message to IVP XML, fully formulates the target SIP URL, and sends it to IVP.
7. The IVP sends a SIP invite to the URL specified by Unified CVP, which will be either the proxy server or Unified CM. The IVP and Unified CM exchange SIP messages, and the caller is connected to the agent via video and/or audio.
8. During the dialog, the agent opts to show a movie to the caller. From either the CAD or CTIOS agent desktop, the agent presses a button on their desktop, which opens a browser to a predetermined URL. The URL had been determined originally by the Unified CVP call server and placed in an ECC variable.
9. The browser connects to the Video Media Server specified in the URL. The agent can search for a list of movies or live videos that match the keyword(s) entered. The agent can either click on a movie to retrieve additional metadata or may elect to preview the movie with a PREVIEW button. This will cause the movie to play locally to the agent (not to the caller). If it is the desired movie, the agent then presses PLAY, which then plays the movie or live video to the caller.
10. The Unified CVP call server receives the PLAY request (via the VMS) with the associated call GUID and instructs IVP via IVP XML messages to play the selected movie on the call. This will result in both the agent and caller hearing and seeing the movie at the same time.
11. While the movie is playing, the caller may elect to stop, pause, resume, forward, or rewind the movie using administrator-configurable DTMF keys. The IVP notifies Unified CVP when a DTMF key is pressed by the caller. If the DTMF key matches one of the configured DTMF tones for VCR control, Unified CVP notifies the IVP to take appropriate action on the movie. Additionally, the agent has the ability to stop, pause, resume, forward, or rewind the movie from buttons on the browser screen. Live video may only be stopped.
12. The agent has the ability to record the conversation using a Start/Stop recording button on their browser window. Whether this will record the caller view, the agent view, or the omni view will be controlled by a global configurable setting in the Operations Console.
13. If the agent puts the caller on hold, Unified CM sends a SIP message to IVP, which in turn plays the preconfigured video-on-hold movie to the caller.

**Caller Requests to be Transferred to a Second Skill Group**

14. If the caller requests to be transferred to a second agent, then the first agent will initiate a transfer from their agent desktop application. This action generates a route request from the PG to the Unified ICM central controller.

15. Unified CM in turn sends SIP re-invite messages to IVP, instructing it to move the caller's media streams. The blind transfer may also be performed as a network transfer via Unified CVP, in which case Unified ICM would instruct Unified CVP to blind-transfer the call. In this case, Unified CVP would send an asynchronous request via IVP XML to IVP, telling it to add the new agent leg to the call.
Distributed Deployments

In a distributed deployment, the ingress gateways are geographically separated from the Unified CVP Call Server. This chapter discusses how these types of deployments should be designed as well as how to handle call survivability and call admission control.

Distributed Gateways

Unified CVP can use several different types of gateways depending on the deployment model. This section discusses each type of gateway and how a distributed deployment can affect them.

Ingress and/or Egress Gateway at the Branch

In this deployment model, ingress gateways located at a branch office are typically used to provide callers with access using local phone numbers rather than centralized or non-geographic numbers. This capability is especially important in international deployments spanning multiple countries. Egress gateways are located at branches either for localized PSTN breakout or for integration of decentralized TDM platforms into the Unified CVP switching solution. Apart from the gateways, all other Unified CVP components are centrally located, and WAN links provide data connectivity from each branch location to the central data center.

Ingress or VoiceXML Gateway at Branch

Consideration needs to be given to other voice services that are being run at the branch. For example, the branch is typically a remote Cisco Unified Communications Manager (Unified CM) site supporting both ACD agent and non-agent phones. This model also implies that the PSTN gateway is used not only for ingress of Unified CVP calls but also for ingress/egress of normal user calls for that site. In circumstances where the VoiceXML and voice gateway functions reside at the same branch location but on separate devices, special attention has to be paid to the dial plan to ensure that the VRU leg is sent to the local VoiceXML resource because the Unified CVP Call Server settransferlabel mechanism applies only to co-resident VoiceXML and voice gateway configurations.
When the ingress gateway and VoiceXML gateway at a branch do not reside on the same gateway, there are two ways to ensure that the calls are handled within the branch and not sent across the WAN to a different VoiceXML gateway:

- **Configure Unified ICM with multiple Customers, one per location.**
  This option relies on the Unified ICM configuration to differentiate between calls based on the Dialed Number. The Dialed Number is associated with a Customer representing the branch site. When a NetworkVRU is needed, the NetworkVRU associated with the Customer in Unified ICM is selected and the caller is sent to that NetworkVRU. This allows you to have multiple NetworkVRUs, each with a unique label. The downside of this method is that each NetworkVRU requires its own VRU scripts in Unified ICM. The Unified ICM configuration overhead of making a change to each NetworkVRU script quickly becomes overwhelming when a large number of remote sites are required. This method is not an option when using a Type10 VRU.

- **Configure Unified CVP using the SigDigits feature.**
  The SigDigits feature in Unified CVP allows you to use the dial plan on the SIP Proxy or gatekeeper to route calls to the correct site. When the call arrives at an ingress gateway, the gateway will prepend digits before sending the call to Unified CVP. Those prepended digits are unique to that site from a dial-plan perspective.

  When the call arrives at Unified CVP, Unified CVP will strip the prepended digits and store them in memory, resulting in the original DID on which the call arrived. Unified CVP then notifies Unified ICM of the call arrival using the original DID, which matches a Dialed Number in Unified ICM.

  When Unified ICM returns a label to Unified CVP in order to transfer the call to a VoiceXML gateway for IVR treatment or to transfer the call to an agent phone, Unified CVP will prepend the digits that it stored in memory before initiating the transfer. The dial plan in the SIP Proxy or gatekeeper must be configured with the prepended digits in such a way to ensure that calls with a certain prepended digit string are sent to specific VoiceXML gateways or egress gateways. The prepended digits are prepended as a tech-prefix when using H.323.

  The term SigDigits is used to describe this feature because the command in Unified CVP to turn on the feature and specify how many significant digits should be stripped is `setsigdigits X` for H.323, and the `sip.properties` file configuration directive is `sip.SigDigits = X`.

  This method is preferred because it involves the least amount of Unified ICM configuration overhead; a single NetworkVRU and single set of VRU scripts and Unified ICM routing scripts is all that is needed. This allows all of the Unified CVP servers and VoiceXML gateways to function as a single network-wide virtual IVR from the perspective of Unified ICM.

  The SigDigits feature can also be used to solve multi-cluster call admission control problems. (See Call Admission Control Considerations, page 3-4, for more information.)

### Co-Located VoiceXML Servers and Gateways

Either all gateways and servers are centralized or each site has its own set of co-located VoiceXML Servers and gateways.

Advantages of co-location:

- A WAN outage does not impact self-service applications.
- No WAN bandwidth is required.
Disadvantages of co-location:
- Extra VoiceXML Servers are required when using replicated branch offices.
- There is additional overhead when deploying applications to multiple VoiceXML servers.

**Gateways at the Branch, with Centralized VoiceXML Servers**

Advantages of centralized VoiceXML:
- Administration and reporting are centralized.
- VoiceXML Server capacity can be shared among branch offices.

Disadvantages of centralized VoiceXML:
- Branch survivability is limited.
- WAN bandwidth must be sized for additional VoiceXML over HTTP traffic.

**Cisco Unified Communications Manager**

In a Unified CVP environment, Cisco Unified Communications Manager (Unified CM) can be an ingress or egress gateway. It is more common for Unified CM to be an egress gateway because typically callers are calling from the PSTN, being queued by Unified CVP, and then being switched to Unified CM for handling by an agent. If the caller is not calling from the PSTN but from an IP phone instead, then Unified CM is an ingress gateway from the perspective of Unified CVP.

**Unified CM as an Egress Gateway**

Unified CVP normally depends on the gatekeeper for dial-plan resolution and call admission control. To deploy Unified CM alongside Unified CVP, you must use Unified CM call admission control for calls between the ingress Unified CVP gateway and the agent IP phone. The ability for Unified CM to identify the ingress Unified CVP gateway correctly is complicated because the Unified CVP Call Server is the component that is actually making the H.323 call to Unified CM. Therefore, Unified CM sees the call as coming from the centralized Unified CVP Call Server rather than from the remote ingress gateway.

The Unified CVP Call Server is able to solve this problem by setting the `sourcesignaladdress` field inside the H.323 setup to the IP address of the ingress gateway. Upon receiving the setup from Unified CVP, Unified CM sees the source signaling address and knows that the address is the one that should be used when determining from which location the call is coming. Because Unified CM has this ingress gateway IP configured, Unified CM will use its Locations call admission control configuration to deduce the bandwidth between the ingress gateway and the destination IP phone locations. The Unified CVP Call Server should not be configured as a gateway in Unified CM; instead, the Unified CVP Call Server should send calls to Unified CM via a gatekeeper-controlled H.323 trunk. (See Call Admission Control Considerations, page 3-4, for more information on call admission control mechanisms.)

**Unified CM as an Ingress Gateway**

When an IP phone initiates a call to Unified CVP, Unified CM acts as the ingress gateway to Unified CVP. An H.323 or SIP trunk is used to send calls to Unified CVP. For more information on these types of call flows, see the chapter on Calls Originated by Cisco Unified Communications Manager, page 6-1.
Call Survivability in Distributed Deployments

Distributed deployments require design considerations for other voice services that are being run at the branch. For example, the branch is typically a remote Unified CM site supporting both ACD agent and non-agent phones. This deployment also implies that the PSTN gateway is used not only for ingress of Unified CVP calls but also for ingress/egress of the regular non-ACD phone calls.

Branch reliability becomes somewhat more of an issue than it is in a centralized Unified CVP model because WANs are typically less reliable than LAN links. Therefore, you must provide mechanisms that are local to the branch to gracefully handle calls that are impacted by loss of a WAN link to the central site.

Call survivability must be considered for both the Unified CVP and non-CVP calls. For the Unified CM endpoint phones, survivability is accomplished via a Cisco IOS feature known as Survivable Remote Site Telephony (SRST). For further details on SRST, refer to the latest version of the Cisco Unified Communications SRND Based on Cisco Unified Communications Manager, available at http://cisco.com/go/srnd

For Unified CVP calls, survivability is handled by a combination of services from a TCL script (survivability.tcl) and SRST functions. The survivability TCL script is used to monitor the H.225 or SIP connection for all calls that ingress through the remote gateway. If a signaling failure occurs, the TCL script takes control of the call and redirects it to a configurable destination. The destination choices for the TCL script are configured as parameters in the Cisco IOS Gateway configuration.

Alternative destinations for this transfer could be another IP destination (including the SRST call agent at the remote site), *8 TNT, or hookflash. With transfers to the SRST call agent at the remote site, the most common target is an SRST alias or a Basic ACD hunt group. For further information about these SRST functions, refer to the Cisco Unified Communications SRND Based on Cisco Unified Communications Manager.

For further information on configuration and application of these transfer methods, refer to the latest version of the Cisco Unified CVP Configuration and Administration Guide, available at http://www.cisco.com.

Call Admission Control Considerations

Call admission control must also be considered from a solution perspective, not just a Unified CVP perspective. These considerations are most evident in the distributed branch office model where there are other voice services, such as Cisco Unified CM, sharing the same gateways with Unified CVP and the amount of bandwidth between the sites is limited. The most important item to consider in this case is which call admission control mechanisms are in place on the network so that a consistent call admission control mechanism can be used to account for all the calls traversing the WAN from that site. If two call admission control mechanisms can admit four calls each and the WAN link is able to handle only four calls, then it is possible for both call admission control entities to admit four calls onto the WAN simultaneously and thereby impair the voice quality. If a single call admission mechanism cannot be implemented, then each call admission control mechanism must have bandwidth allocated to it. This situation is not desirable because it leads to inefficient bandwidth over-provisioning.

There are three call admission control mechanisms that can be used in a Unified CVP environment: gatekeeper call admission control, Unified CM Locations, and Unified CM RSVP Agent. In a single-site deployment, call admission control is not necessary.
Unified CM performs call admission by assigning devices to certain *locations* and keeping track of how many calls are active between these locations. Because Unified CM knows how many calls are active and what codec is in use for each call, it is able to calculate how much bandwidth is in use and to limit the number of calls allowed.

A thorough conceptual understanding of call admission control mechanisms is important. These mechanisms are explained in the *Cisco Unified Communications SRND Based on Cisco Unified Communications Manager*, available at http://cisco.com/go/srnd

### Gatekeeper Call Admission Control

In a pure TDM environment where Unified CVP is switching calls from an ingress gateway to an egress gateway attached to a TDM ACD/IVR, the gatekeeper can handle the call admission control functionality.

If Unified CM is the egress gateway, gatekeeper call admission control can be used only if the ingress Unified CVP gateways and the IP phones are at different sites. Note that gatekeeper dial-plan resolution is still in use.

Because Unified CM locations-based call admission control is used between the remote sites of a cluster, a gatekeeper typically is used for dial-plan resolution only. Understanding the routing of calls in the dial plan and the gatekeeper resolution is important because call routing situations might occur in which it is necessary to use more than one set of gatekeepers in the implementation. This is particularly common when using this model in a situation where more than one Unified CM cluster are being used to control the remote sites. For further discussion and information on this topic, see *H.323 Gatekeeper Call Routing*, page 3-9.

### Unified CM Call Admission Control

If Unified CM is sending or receiving calls from Unified CVP and there are Unified CVP gateways and IP phone agents co-located at remote sites, it is important to understand the call flows in order to design and configure call admission control correctly.

### H.323 Call Flows

The gatekeeper and Unified CM do not share bandwidth usage information. Networks shared by both the gatekeeper and IP phones will have two separate call admission control mechanisms determining if there is enough bandwidth to place a call. Instead of using the gatekeeper for call admission control, it is possible to use Unified CM Locations as the call admission control mechanism for Unified CVP calls. How Unified CM determines an endpoint's location is key to designing call admission control properly.

Consider the basic call flow of a Unified CVP call versus a non-CVP call. When a user picks up an IP phone and makes a call from the remote site to the central site, Unified CM considers the location definitions of the endpoints and the codec requirements defined in the Unified CM Region configurations and decides whether or not to allow the call. Note that the call admission control and the codec requirements are controlled between these endpoints by Unified CM as the controlling call agent.

By default, Unified CM looks at the source IP address of an incoming H.323 call to determine which H.323 device is originating the call. Unified CM then uses the configuration of this device to determine its location and to perform call admission control for the call. When Unified CVP is delivering calls from a remote branch gateway to a Unified CM IP Phone, Unified CVP is in the middle of the H.323 signaling,
so the source IP address from Unified CM's perspective is the Unified CVP Server. Because the Unified CVP Server is centralized along with Unified CM, it is not possible to perform call admission control based on the Unified CVP Server's IP address. Unified CM must be aware that calls arriving from Unified CVP are coming from a gateway at a specific branch so that it can calculate call admission control correctly. In order to solve this problem, Unified CVP must be configured to insert information in the payload of the H.323 SETUP message that identifies the IP address of the originating gateway, and Unified CM must be configured to look at this information when determining on which gateway an H.323 call is arriving.

This requires enabling one Unified CM service parameter and ensuring that another parameter is set to its default value:

- Change the cluster-wide service parameter **Accept unknown TCP connection** to True. (The default is False.)
- Ensure that the service parameter **Device Name of gatekeeper trunk that will use port 1720** remains at its default setting of blank.

When set to True, the service parameter Accept Unknown TCP Connection changes the behavior for inbound H.323 calls. Unified CM accepts an unknown H.225 TCP connection and waits for the H.323 SETUP message. Unified CM then extracts the User-to-User Information Element (UUIE) and examines the sourceCallSignalAddress field, which contains the IP address of the originating gateway. Unified CM compares this address against its configured gateways. If a match is found, the call is treated as if it originated from the voice gateway and not the Unified CVP Server. The Unified CVP Server IP address must not be configured as an H.323 gateway, otherwise Unified CM will match first on the source IP address and will not look at the information in the sourceCallSignalAddress field. In order to deliver calls to Unified CM from Unified CVP without specifying Unified CVP as an H.323 gateway, you must configure an H.323 Gatekeeper Trunk in Unified CM so that Unified CVP will sends calls to Unified CM via the gatekeeper over the trunk.

The Unified CM service parameter **Device Name of gatekeeper trunk that will use port 1720** is used to force a gatekeeper-controlled trunk to register to a gatekeeper on port 1720. This feature causes any inbound H323 call that is signaling on port 1720 to be treated immediately as a gatekeeper-controlled trunk call. The H.225 signaling address is not examined in this case.

This behavior is not how Unified CM traditionally treats a gatekeeper-controlled H.323 call. Typically, all gatekeeper-controlled calls come from the hub location (location None or Hub_None). These changes ensure that the call is not treated as a gatekeeper-controlled call and that locations-based call admission control is applied. Note that, in this model, if Unified CM does not match the gateway signaling address in its list of configured gateways, it rejects the call. Figure 3-1 illustrates the decision tree for H.323 call processing.
To configure Unified CVP to work correctly with Unified CM call admission control, use VBAdmin to set the Unified CVP Server parameter `setlocationsbasedcac on`. This setting tells Unified CVP to populate the sourceCallSignalAddress field and to use TCP port 1720 when sending calls to Unified CM.

When using this feature in conjunction with calls originated by Unified CM, the sourceCallSignalAddress populated by Unified CVP will be the IP address of Unified CM. If the call is transferred back to Unified CM, it will inspect this field and try to find a configured gateway with that IP address, but the call will fail because normally Unified CM will never be configured as a gateway. As a workaround to this problem, configure each Unified CM in the cluster as an H.323 gateway, but be sure to never configure the Unified CM dial plan to send calls using those gateways.

### Multiple Cisco Unified CM Clusters

When more than one centralized Unified CM cluster are used for the remote sites, additional consideration must be given to routing calls based on agent selection. In a multi-cluster environment, each cluster manages a group of remote sites and tracks the voice calls to those sites within the locations-based call admission control mechanism. Using the changes discussed above, Unified CM considers H.323 calls within its locations-based call admission control mechanism. Because H.323 is a peer-to-peer protocol, an H.323 gateway can signal a call to any other call agent that will accept it. Considering the locations-based call admission control mechanism described above, it is necessary for the remote gateway to be told to signal a call to the Unified CM cluster that owns the location of that
remote gateway. However, in a Unified CCE environment, Unified ICM tracks the availability of agents without considering on which cluster they are located. This ability provides great scalability for Unified CCE but must be accounted for in this type of implementation.

If a call coming in through a Unified CVP ingress gateway is destined for an IP Phone at a different site registered to a different cluster, the call must first be routed through the cluster that is handling call admission control for the ingress gateway, and then routed to the destination cluster. If the call is routed directly from the ingress gateway to the destination cluster, the cluster that is handling call admission control for the ingress gateway is not aware of the call traversing the WAN and does not deduct bandwidth appropriately.

This call routing can be handled by using the SigDigits feature in Unified CVP and its associated dial-plan configuration. The SigDigits feature in Unified CVP allows you to use the dial plan on the SIP Proxy or gatekeeper to route calls to a specific Unified CM cluster. When the call arrives at an ingress gateway, the gateway will prepend digits before sending the call to Unified CVP. Those prepended digits are unique to that site from a dial-plan perspective. When the call arrives at Unified CVP, Unified CVP will strip the prepended digits and store them in memory, resulting in the original DID on which the call arrived. When Unified ICM returns a label to Unified CVP in order to transfer the call to an agent phone, Unified CVP will prepend the digits that it stored in memory before initiating the transfer. The dial-plan configuration in the SIP Proxy or gatekeeper is configured with the prepended digits so that calls with a certain prepended digit string are sent to a specific Unified CM cluster. The digits are prepended as a tech-prefix when using H.323.

For more information on how the SigDigits feature works, see Distributed VoiceXML Gateways (Separate Ingress Gateway and VoiceXML), page 4-19.

### SIP Call Flows

With SIP-based call flows, Cisco Unified CM 6.0 (and prior releases) is able to look at only the source IP address of the incoming SIP INVITE from Unified CVP. This limitation causes a problem with call admission control because Unified CM is not able to identify which gateway behind Unified CVP originated the call.

Cisco Unified CM 6.1 introduces a new feature that functions in a similar manner to the H.323 call flow. Cisco Unified CM 6.1 has enhanced the SIP Trunk to look beyond the source IP address and to inspect information contained in the SIP header when determining which device originated a call. More specifically, the Call-Info header in the SIP INVITE will specify the originating device in the following format:

```
<sip: IPAddress:port>;purpose=x-cisco-origIP
```

Where `IPAddress:port` indicates the originating device and its SIP signaling port.

By default, a fresh installation of Unified CVP 7.0 and later releases will populate this field. However, with upgrades from Unified CVP 4.x, Unified CVP does not populate the Call-Info header with this field. In order to configure Unified CVP to insert this information into the header, you must edit the sip.properties file to include the following line:

```
SIP.UseCCMCAC=true
```

When using SIP with Cisco Unified CM 6.0 or prior releases and Unified CVP deployments with remote gateways and IP phones, there will not be a fully integrated call admission control mechanism, therefore you must use one of the following mechanisms instead:

- Use H.323 as the signaling protocol.
- Deploy a Unified CVP Server at each site.
- Over-provision enough voice priority queue bandwidth so that call admission control is not needed.
Note that this limitation is not an issue for deployments that do not need call admission control, such as a campus LAN or MAN, where all of the devices are connected with high-bandwidth links.

**RSVP**

Cisco Unified CM 5.0 introduced support for Resource Reservation Protocol (RSVP) between endpoints within a cluster. RSVP is a protocol used for call admission control and is used by the routers in the network to reserve bandwidth for calls. RSVP can be used for delivering calls to Unified CCE agents in a Unified CM cluster. Unified CM uses the RSVP Agent associated with each device in order to complete the RSVP reservation. Because Unified CM must be able to identify the originating gateway, you must use the same configuration of Unified CM for locations-based call admission control with Unified CVP. This configuration enables Unified CM to correctly identify the remote gateway and its associated RSVP agent and to complete the RSVP reservation. Because Unified CM is currently not able to identify SIP gateways behind Unified CVP, RSVP has the same limitation as locations-based call admission control with SIP.

For more information on RSVP, refer to the latest version of the *Cisco Unified Communications SRND Based on Cisco Unified Communications Manager*, available at [http://cisco.com/go/srnd](http://cisco.com/go/srnd)

**H.323 Gatekeeper Call Routing**

For proper configuration of remote H.323 gateways with a Unified CM cluster, first consider the H.225 implications of this configuration without the use of gatekeeper.

When configuring dial-peer destinations for the Cisco IOS Gateways, you must configure a dial peer pointing to the IP addresses of the Unified CM servers that are processing calls for that gateway. These server IP addresses must be the same servers that are in the redundancy group of the device pool definition for that gateway in the Unified CM configuration. (See Figure 3-2.) If the remote H.323 gateway sends a call to a Unified CM server that is not in the redundancy group for that gateway, the call is rejected. For example, if the Madison gateway in Figure 3-2 sends a call to the.3 server, the call is rejected.
Call Admission Control Considerations

Chapter 3: Distributed Deployments

Figure 3-2  Dial Peer Configuration

While the example in Figure 3-2 is simple to understand, the configuration can become challenging to maintain for several hundred remote sites over an extended period of time. If the Cisco Unified CM gatekeeper redundancy group is changed, all the remote H.323 gateway dial-peer targets must be changed to match the new IP address of the server added to the redundancy group. A gatekeeper can help reduce this challenge.

When using the gatekeeper for configuration, the H.323 gateway makes a Registration Admission Status (RAS) request to the gatekeeper for an IP address to which to send the call. The gatekeeper automatically responds with one of the Unified CM server addresses defined in the redundancy group for the gatekeeper trunk. If the redundancy group is changed, Unified CM must re-register to the gatekeeper. However, no further configuration is necessary on the remote gateway.

- Be sure to configure a dial peer for each Cisco Unified CallManager server in the redundancy group and device pool assigned to the gateway in Cisco Unified CallManager.
- Ensure that the IP addresses in the configurations match on both sides of the link.
- The Cisco Unified CallManager redundancy group for the Madison gateway contains the .1 and .2 servers.

Preparation

dial-peer voice 1 voip
destination-pattern 1...
preference 1
session target ipv4:10.10.10.1
dial-peer voice 2 voip
destination-pattern 1...
preference 2
session target ipv4:10.10.10.2

Publisher

San Jose Headquarters

WAN

Madison

Dial Peer Configuration

191325

.1.3

.2

.1

.2
Designing Unified CVP for High Availability

This chapter describes guidelines and best practices for designing a high-availability Unified CVP system.

Overview

A high-availability design provides the highest level of failure protection. Your solution may vary depending upon business needs such as:

- Tolerance for call failure
- Budget
- Topological considerations
Unified CVP can be deployed in many configurations that use numerous hardware and software components. Each solution must be designed in such a way that a failure impacts the fewest resources in the call center. The type and number of resources impacted depends on how stringent the business requirements are and which design characteristics you choose for the various Unified CVP components, including the network infrastructure. A good Unified CVP design is tolerant of most failures (defined later in this chapter), but sometimes not all failures can be made transparent to the caller.

Unified CVP is a sophisticated solution designed for mission-critical call centers. The success of any Unified CVP deployment requires a team with experience in data and voice internetworking, system administration, and Unified CVP application configuration.

Before implementing Unified CVP, use careful preparation and design planning to avoid costly upgrades or maintenance later in the deployment cycle. Always design for the worst possible failure scenario, with future scalability in mind for all Unified CVP sites.

In summary, plan ahead and follow all the design guidelines and recommendations presented in this guide and in the latest version of the *Cisco Unified Communications Solution Reference Network Design (SRND) Based on Cisco Unified Communications Manager*, available at:

   http://www.cisco.com/go/srnd

For assistance in planning and designing your Unified CVP solution, consult your Cisco or certified Partner Systems Engineer (SE).

**A Note About the Unified CVP Call Server Component**

The other chapters of this document treat the Unified CVP Call Server as a single component because those chapters have no need to examine it in any more depth than that. When discussing Unified CVP high availability however, it is important to understand that there are actually several parts to this component:

- **H.323 Service** — Responsible for H.323 processing of incoming and outgoing calls as well as registering with the gatekeeper. The H.323 Service was known as the Unified CVP Voice Browser in previous versions of Unified CVP.
- **SIP Service** — Responsible for processing incoming and outgoing calls via SIP.
- **ICM Service** — Responsible for the interface to ICM. The ICM Service communicates with the VRU PG using GED-125 to provide ICM with IVR control. The ICM Service was part of the Application Server in previous releases of Unified CVP, but now it is a separate component.
- **IVR Service** — Responsible for the conversion of Unified CVP Microapplications to VoiceXML pages, and vice versa. The IVR Service was known as the Application Server in previous Unified CVP versions.
Layer 2 Switch

Figure 4-1 shows a high-level layout for a fault-tolerant Unified CVP system. Each component in the Unified CVP site is duplicated for redundancy. The quantity of each of these components varies based on the expected busy hour call attempts (BHCA) for a particular deployment. The following sections describe the failover strategy for each of these components.

In Figure 4-1, two switches provide the first level of network redundancy for the Unified CVP Servers:

- If one switch fails, only a subset of the components becomes inaccessible. The components connected to the remaining switch can still be accessed for call processing.
- If a Content Services Switch (CSS) is used, its redundant partner must reside on the same VLAN in order to send keep-alive messages to each other via Virtual Router Redundancy Protocol (VRRP), a protocol similar to Hot Standby Router Protocol (HSRP). If one of the switches fails, the other CSS is still functional.

For more information on data center network design, refer to the Data Center documentation available at http://www.cisco.com/go/srnd

---

**Note**

NIC teaming is not currently supported in the Unified CVP solution.

---

**Note**

Cisco recommends that the NIC card and ethernet switch be set to 100 MB full duplex for 10/100 links, or set to auto-negotiate for gigabit links.
Originating Gateway

The function of the originating gateway in a Unified CVP solution is to accept calls from the PSTN and direct them to Unified CVP for call routing and IVR treatment.

This section covers the following topics:

- Configuration, page 4-4
- Call Disposition, page 4-13

Configuration

For the most current information on how to provide redundancy and reliability for originating gateways and T1/E1 lines, refer to the latest version of the Cisco Unified Contact Center Enterprise Solution Reference Network Design (SRND), available at http://www.cisco.com/go/srnd

In addition, consider the following issues when designing gateways for high availability in a Unified CVP solution:

- When used in ICM-integrated models, the originating gateway communicates with Unified CVP using H.323 or SIP. Unlike MGCP, SIP and H.323 do not have redundancy features built into the protocol. Instead, SIP and H.323 rely on the gateways and call processing components for redundancy.

- When configuring the gateway, it is best to bind the H.323 or SIP signaling to the virtual loopback interface, as illustrated in the following configuration examples:

  H.323:
  ```
  interface Loopback0
  ip address 10.0.0.10 255.255.255.255
  h323-gateway voip interface
  h323-gateway voip id sj-gk ipaddr 10.0.1.100 1719 <- GK IP
  h323-gateway voip h323-id sj-gw1
  h323-gateway voip bind srcaddr 10.0.0.10
  ```

  SIP:
  ```
  voice service voip
  sip
  bind control source-interface Loopback0
  bind media source-interface Loopback0
  ```

  This configuration allows call signaling to operate independent of the physical interfaces. In this way, if one interface fails, the other interface can handle the traffic. Each gateway interface should be connected to a different physical switch to provide redundancy in the event that one switch or interface fails. Each interface on the gateway is configured with an IP address on a different subnet. The IP Router(s) for the network is then configured with redundant routes to the Loopback address through the use of static routes or a routing protocol. If a routing protocol is used, pay careful attention to the number of routes being exchanged with the gateway, and consider using filters to limit the routing updates so that the gateway is only advertising the loopback address and not receiving routes.
Call Disposition

If the originating gateway fails, the following conditions apply to call disposition:

- Calls in progress are dropped. There is nothing that can be done to preserve these calls because the PSTN switch has lost the D-channel to all T1/E1 trunks on this gateway.
- New calls are directed by the PSTN carrier to a T1/E1 at an alternate gateway, provided the PSTN switch has its trunks and dial plan configured to do so.

SIP Proxy

A SIP Proxy server plays a similar role to the gatekeeper in a Unified CVP solution. The SIP Proxy server provides dial plan resolution on behalf of SIP endpoints, allowing dial plan information to be configured in a central place instead of statically on each SIP device. A SIP Proxy server is not required in a Unified CVP solution, but it is used in most solutions because of the benefits of centralized configuration and maintenance. Multiple SIP Proxy servers can be deployed in the network to provide load balancing, redundancy, and regional SIP call routing services. In a Unified CVP solution, the choices for SIP call routing are:

- SIP Proxy Server
  - Advantages:
    - Weighted load balancing and redundancy.
    - Centralized dial-plan configuration.
    - SIP Proxy may already exist or be used for other applications for dial-plan resolution or intercluster call routing.
  - Disadvantages:
    - Additional server and/or hardware required for SIP Proxy if not already deployed.

- Static routes using DNS SRV records on a DNS Server
  - Advantages:
    - Weighted load balancing and redundancy.
  - Disadvantages:
    - Might not be able to use an existing DNS server, depending on the location of the DNS server. The ability to share or delegate DNS server administration rights might not be possible in some organizations.
    - Dial-plan configuration needs to be configured on each device individually (Cisco Unified Communications Manager, Unified CVP, and gateways).
    - DNS SRV lookup is performed for each and every call by Unified CVP. If the DNS server is slow to respond, is unavailable, is across the WAN, or so forth, this will affect performance.

- Static routes using local DNS SRV records
  - Advantages:
    - Weighted load balancing and redundancy.
    - Does not depend on an external DNS Server, thus eliminating a point of failure, latency, and DNS Server performance concerns.
Disadvantages:
Dial plan must be configured on each device individually (Cisco Unified Communications Manager, Unified CVP, and gateways).

Static routes using IP addresses

- Advantages:
  Does not depend on any other device (DNS or Proxy) to deliver calls to the destination.

- Disadvantages:
  No redundancy possible for SIP calls from Unified CVP.
  Dial plan must be configured on each device individually.

This option makes sense only for environments that do not have redundancy (single server) or for lab deployments.

Each device in the Unified CVP solution can use the above methods for determining where to send a call. The Unified CVP Call Server interface to the SIP network is through the Unified CVP SIP Service, which is discussed in the section on Unified CVP SIP Service, page 4-8.

Configuration

The following sections discuss configuration of the SIP Proxy Server and Cisco IOS Gateways using SIP. It is not meant to be an exhaustive list of configuration options but only highlights certain configuration concepts.

SIP Proxy Server Configuration

The SIP Proxy Server should be configured with static routes that point at the appropriate devices (Call Servers, VoiceXML gateways, Cisco Unified Communications Manager cluster, and so forth). The SIP Proxy Server configuration allows you to specify the priority of the routes. In the case where there are multiple routes to the same destination, you can configure the SIP Proxy to load-balance across the destinations with equal priority or to send the calls in a prioritized manner using different priorities.

The Cisco Unified Presence Server SIP Proxy cannot use DNS SRV for outbound calls; it must be configured with multiple static routes in order to do load balancing and failover. The static routes can point to an IP address or a regular DNS A host record.

To reduce the impact of a Proxy Server failure, Cisco recommends that you disable the RecordRoute header from being populated by the SIP Proxy Server. In this way, the inbound calls route through a SIP Proxy; but once they reach the Unified CVP Call Server, the signaling is exchanged directly between the originating device and the Call Server, and a SIP Proxy failure will not affect the calls in progress.

Cisco also highly recommends using UDP instead of TCP for SIP signaling. TCP stack timeout delays can cause significant delays to the caller during failures.

Cisco IOS Gateway Configuration

With Cisco IOS gateways, dial-peers are used to match phone numbers, and the destination can be a SIP Proxy Server, DNS SRV, or IP address. The following example shows a Cisco IOS gateway configuration to send calls to a SIP Proxy Server using the SIP Proxy's IP address.

```
sip-ua
sip-server ipv4:10.4.1.100:5060
```
The **sip-server** command on the dial-peer tells the Cisco IOS gateway to use the globally defined sip-server that is configured under the **sip-ua** settings. In order to configure multiple SIP Proxies for redundancy, you can change the IP address to a DNS SRV record, as shown in the following example. The DNS SRV record allows a single DNS name to be mapped to multiple servers.

```plaintext
sip-ua
sip-server dns:cvp.cisco.com
```

Alternatively, you can configure multiple dial-peers to point directly at multiple SIP Proxy servers, as shown in the following example. This configuration allows you to specify IP addresses instead of relying on DNS.

```plaintext
dial-peer voice 1000 voip
session target ipv4:10.4.1.100
preference 1
...
dial-peer voice 1000 voip
session target ipv4:10.4.1.101
preference 1
...
```

In the preceding examples, the calls are sent to the SIP Proxy server for dial plan resolution and call routing. If there are multiple Unified CVP Call Servers, the SIP Proxy server would be configured with multiple routes for load balancing and redundancy. It is possible for Cisco IOS gateways to provide load balancing and redundancy without a SIP Proxy Server. The following example shows a Cisco IOS gateway configuration with multiple dial-peers so that the calls are load-balanced across three Unified CVP Call Servers.

```plaintext
dial-peer voice 1001 voip
session target ipv4:10.4.33.131
preference 1
...
dial-peer voice 1002 voip
session target ipv4:10.4.33.132
preference 1
...
dial-peer voice 1003 voip
session target ipv4:10.4.33.133
preference 1
...
```

DNS SRV records allow an administrator to configure redundancy and load balancing with finer granularity than with DNS round-robin redundancy and load balancing. A DNS SRV record allows you to define which hosts should be used for a particular service (the service in this case is SIP), and it allows you to define the load-balancing characteristics among those hosts. In the following example, the redundancy provided by the three dial-peers configured above is replaced with a single dial-peer using a DNS SRV record. Note that a DNS server is required in order to do the DNS lookups.

```plaintext
ip name-server 10.4.33.200

...  
dial-peer voice 1000 voip
session target dns:cvp.cisco.com
```
With Cisco IOS gateways, it is possible to define DNS SRV records statically, similar to static host records. This capability allows you to simplify dial-peer configuration while also providing DNS SRV load balancing and redundancy. The downside of this method is that, if the SRV record needs to be changed, it must be changed on each gateway instead of on a centralized DNS server. The following example shows the configuration of static SRV records for SIP services handled by cvp.cisco.com, and the SIP SRV records for cvp.cisco.com are configured to load-balance across three servers.

```text
ip host cvp4cc2.cisco.com 10.4.33.132
ip host cvp4cc3.cisco.com 10.4.33.133
ip host cvp4cc1.cisco.com 10.4.33.131

(SRV records for SIP/TCP)
ip host _sip._tcp.cvp.cisco.com srv 1 50 5060 cvp4cc3.cisco.com
ip host _sip._tcp.cvp.cisco.com srv 1 50 5060 cvp4cc2.cisco.com
ip host _sip._tcp.cvp.cisco.com srv 1 50 5060 cvp4cc1.cisco.com

(SRV records for SIP/UDP)
ip host _sip._udp.cvp.cisco.com srv 1 50 5060 cvp4cc3.cisco.com
ip host _sip._udp.cvp.cisco.com srv 1 50 5060 cvp4cc2.cisco.com
ip host _sip._udp.cvp.cisco.com srv 1 50 5060 cvp4cc1.cisco.com
```

Cisco highly recommends the use of UDP instead of TCP for SIP signaling. TCP stack timeout delays can cause significant delays to the caller during failures.

**Call Disposition**

Calls are handled as indicated for the following failure scenarios:

- Primary SIP Proxy Server fails
  
  Active calls are preserved. Subsequent transfers of calls are successful, provided the backup SIP Proxy is available and the RecordRoute header is not being populated by the SIP Proxy. If the RecordRoute header is populated, signaling to the gateway will not be possible and subsequent transfer attempts will fail.

- All SIP Proxy Servers fail or are unreachable
  
  New calls arriving at the gateway are default-routed if survivability is configured on the gateway.

**Unified CVP SIP Service**

The Unified CVP SIP Service is the service on the Unified CVP Call Server that handles all incoming and outgoing SIP messaging and SIP routing. The Unified CVP SIP Service can be configured to use a SIP Proxy server for outbound dial plan resolution, or it can be configured to use static routes based on IP address or DNS SRV. Unified CVP Call Servers do not share configuration information about static routes; therefore, if a change needs to be made to a static route, then the change must be made on each Call Server's SIP Service. Cisco recommends that you use a SIP Proxy Server to minimize configuration overhead.
Configuration

If only a single SIP Proxy server is needed for outbound call routing from the Unified CVP Call Server, choose the SIP Proxy configuration when configuring the SIP Service. In the Operations Console, configure the following:

- Add a SIP Proxy Server and specify the IP address of the server.

Under the Call Server SIP Service settings, configure the following:

- Enable Outbound Proxy = True
- Use DNS SRV type query = False
- Outbound Proxy Host = SIP Proxy Server configured above

When using multiple SIP Proxy servers for outbound redundancy from the Call Server, configure the SIP Proxy with a DNS name and configure DNS SRV records in order to reach the SIP Proxy Servers. The DNS SRV records can exist on an external DNS Server, or they can be configured in a local DNS SRV record on each CVP server. In the OAMP Console, configure the following:

- Add a SIP Proxy Server and specify DNS name of the server.

Under the SIP Service configuration, configure the following:

- Enable Outbound Proxy = True
- Use DNS SRV type query = True
- The DNS SRV record should then be configured with the list of SIP Proxy Servers.

Cisco also highly recommends using UDP instead of TCP for SIP signaling. TCP stack timeout delays can cause significant delays to the caller during failures.

To configure the Local DNS SRV record on each server, perform the following steps:

1. Create a file named srv.xml in the c:\cisco\cvp\conf directory, with the following format:

   ```xml
   <locater>
     <host name="vxml.cisco.com">
       <record weight="1" priority="1" destination="10.4.33.1" port="5060"/>
       <record weight="1" priority="1" destination="10.4.33.2" port="5060"/>
     </host>
     <host name="cm.cisco.com">
       <record weight="1" priority="1" destination="10.4.33.101" port="5060"/>
       <record weight="1" priority="2" destination="10.4.33.100" port="5060"/>
     </host>
   </locater>
   ```

2. Under the SIP Service configuration, check Resolve SRV records locally.

Configuring High Availability for Calls in Progress

In the event that a Unified CVP Call Server fails with calls in progress, it is possible to salvage all calls if certain gateway configuration steps have been taken. A Call Server can fail in one of several ways:

- The server can crash.
- The process can crash.
- The process can hang.
- There can be a network outage.
The configuration discussed in this section protects against all of these situations. However, the following two situations cannot be protected against:

- Someone stops the process with calls in progress. This situation occurs when a system administrator forgets to put the Call Server out-of-service first to allow calls in progress to finish before stopping the process.

- The Call Server exceeds the recommended call rate. Although there is a throttle for the absolute number of calls allowed in the Call Server, there is no throttle for call rate. In general, exceeding 5 calls per second (cps) for an extended period of time can cause erratic and unpredictable call behavior on certain components of the CVP solution if one of the components is not sized correctly or if the call load is not balanced according to the weight and sizing of each call processing component.


The survivability.tcl script itself also contains some directions and useful information.

In the event of most downstream failures (including a Unified CVP Call Server failure), the call is default-routed by the originating gateway. Note that survivability is not applicable in the Unified CVP Standalone and NIC-routing models because there is no Unified CVP H.323 or SIP Service involved anywhere in those models.

There is also a mechanism for detection of calls that have been cleared without Unified CVP's knowledge:

- Unified CVP checks every 2 minutes for inbound calls that have a duration older than a configured time (the default is 10 minutes).

- For those calls, Unified CVP sends an UPDATE message. If the message receives a rejection or is undeliverable, then the call is cleared and the license released.

The CVP SIP Service can also add the Session Expires header on calls so that endpoints such as the originating gateway may perform session refreshing on their own. RFC 4028 (Session Timers in the Session Initiation Protocol) has more details on the usage of Session Expires with SIP calls.

**Call Disposition**

Calls are handled as indicated for the following scenarios:

- Calls in progress

  If the Unified CVP SIP Service fails after the caller has been transferred (transfers include transfer to an IP phone, VoiceXML gateway, or other egress gateway), then the call continues normally until a subsequent transfer activity (if applicable) is required from the Unified CVP SIP Service. If the caller has not hung up and is awaiting further activity, there is a period of 9 to 18 seconds of silence before the caller is default-routed by survivability to an alternate location.

  If the call has not yet been transferred, the caller hears 9 to 18 seconds of silence before being default-routed by survivability to an alternate location. (Survivability does not apply in the Unified CVP Standalone and NIC-routing models.)
• New calls

New calls are directed by the SIP Proxy to an alternate Unified CVP Call Server. If no Call Servers are available, the call is default-routed to an alternate location by survivability. (Survivability does not apply in the Unified CVP Standalone and NIC-routing models.)

Gatekeeper

An H.323 gatekeeper is required when using H.323 in all ICM-integrated deployment models except Model #4 (VRU Only with NIC Controlled Routing), which does not use Unified CVP for call control at all. Additionally, if SIP is used as the call control protocol, the gatekeeper is not required. The Unified CVP H.323 Service requires the use of a gatekeeper, and Unified CVP H.323 Service is used in all of these other models. Note, however, that a gatekeeper is an optional (although recommended) component for call routing by the originating gateway in those deployments. An originating gateway can perform all of its H.323 call routing by using VoIP dial-peers that contain static IP addresses, whereas the Unified CVP H.323 Service must always perform a gatekeeper Remote Access Service (RAS) lookup to route calls.

Note

In one particular situation, when using the VBAdmin SetTransferLabel option, the H.323 Service ignores the IP address returned from the gatekeeper and instead routes the IVR call leg back to the originating gateway from which the call arrived. This feature ensures that no WAN bandwidth is used during IVR treatment or queuing. A gatekeeper is still required in this situation because the H.323 Service has to perform the gatekeeper lookup function to obtain possible alternate endpoints in the event that the attempt to transfer the call to the originating gateway fails.

Unified CVP can use one of the following types of gatekeeper high-availability mechanisms:

• Gatekeeper Redundancy Using HSRP, page 4-11
• Gatekeeper Redundancy Using Alternate Gatekeeper, page 4-12

Only HSRP and alternate gatekeeper are supported by Unified CVP. Alternate gatekeeper support was introduced in Unified CVP 3.1 SR1.

Gatekeeper Redundancy Using HSRP

HSRP is a Cisco proprietary router redundancy protocol that allows two or more gatekeepers to share the same IP address in an active/standby fashion. Using HSRP, two gatekeepers work together to present the appearance of a single virtual IP address on the LAN.

The gatekeepers share the same IP and MAC addresses. Therefore, if one of the gatekeepers fails, the hosts on the LAN are able to continue forwarding packets to a consistent IP and MAC address. The process of transferring the routing responsibilities from one device to another is transparent to the user. The H.323 endpoints (such as the Unified CVP H.323 Service, Cisco Unified Communications Manager, and gateways) register to a virtual IP address that represents the HSRP gatekeeper pair.

If one gatekeeper fails, its partner assumes primary control. The major disadvantage of HSRP is that both gatekeepers in the HSRP failover pair must reside on the same IP subnet or VLAN, therefore they generally cannot be separated geographically. Gatekeepers using HSRP for redundancy also do not share any state information. Therefore, when a failover occurs, all of the devices must re-register with the gatekeeper from scratch.

As of Unified CVP 3.1 SR1, HSRP is no longer recommended. Instead gatekeeper clustering and alternate gatekeeper configuration on Unified CVP is the preferred method of gatekeeper redundancy.
Gatekeeper Redundancy Using Alternate Gatekeeper

The Unified CVP H.323 Service can be configured with a list of alternate gatekeepers (as many as needed; there is no limit). When the H.323 Service starts, it attempts to register to the first gatekeeper in the list. If the registration is not successful, it tries the remaining gatekeepers sequentially in the list until a successful registration occurs.

The H.323 Service stays registered to that gatekeeper until either of the following events occurs:

- That gatekeeper has some type of failure. The H.323 Service recognizes a gatekeeper failure in the following ways:
  - The periodic RAS Registration Request (RRQ) to the gatekeeper times out or is rejected.
  - An Admission Request (ARQ) on a transfer times out.
  - The gatekeeper pro-actively tells the H.323 Service to unregister, such as when the administrator does a shutdown on the gatekeeper configuration.
- The user does another setGK from VBAadmin. This causes the H.323 Service to register with the first gatekeeper in the list, if that gatekeeper is available; otherwise, it once again does a sequential attempt.

Gatekeeper clustering is not required in order to use Unified CVP alternate gatekeeper. It is possible to have two gatekeepers identically configured and also configure Unified CVP with alternate gatekeepers to provide redundancy.

The Unified CVP H.323 Service does not support gatekeeper clustering messages, but there is no reason that the gatekeepers cannot be part of a GUP cluster. In this way, other H.323 endpoints that natively support clustering (such as Cisco Unified Communications Manager and Cisco IOS gateways) can take advantage of the benefits of gatekeeper clustering. Unified CVP simply ignores clustering messages, such as when one of the gatekeepers in the cluster becomes overloaded or when Unified CVP registers with the gatekeeper.

Because Unified CVP does not automatically learn the other members of the gatekeeper cluster when it registers to the gatekeeper, it is necessary to define the gatekeeper cluster members statically in Unified CVP. Unified CVP uses one or more of the gatekeepers in the cluster as the alternate gatekeepers in its list and detects failure according to the rules mentioned earlier in this section.

Configuration

This section covers the following topics:

- HSRP Configuration, page 4-12
- Alternate Gatekeeper, page 4-13

HSRP Configuration

On the primary gatekeeper, enter these commands:

```
interface ethernet 0
ip address 10.0.1.98 255.255.255.0
! Unique IP address for this GK
standby 1 ip 10.0.1.100
! Member of standby group 1, sharing virtual address 10.0.1.100
standby 1 preempt
! Claim active role when it has higher priority.
standby 1 priority 110
! Priority is 110.
```
On the backup gatekeeper, enter these commands:

```bash
interface ethernet 0
ip address 10.0.1.99 255.255.255.0
standby 1 ip 10.0.1.100
standby 1 preempt
standby 1 priority 100
```

On both gatekeepers, enter identical gatekeeper configurations. For example:

```bash
gatekeeper
! Enter gatekeeper configuration mode.
zone local gk-sj cisco.com 10.0.1.100
! Define local zone using HSRP virtual address as gatekeeper RAS address.
```

For more information, refer to the latest version of the *Configuration and Administration Guide for Cisco Unified Customer Voice Portal (CVP)*, available at


## Alternate Gatekeeper

Configure alternate gatekeepers using Unified CVP VBAdmin, as shown in the following examples:

```bash
set GK "10.0.1.100, 10.0.2.100, 10.0.3.100"
```

This example sets up three gatekeepers to which the H.323 Service could possible register. In each case, the H.323 Service registers to the first local zone that is configured in that gatekeeper. It also uses the default RAS port 1719.

```bash
setGK "10.0.1.100:zone1:1718, 10.0.2.100"
```

This example causes the H.323 Service to attempt to register first to gatekeeper 10.0.1.100 on port 1718 with local zone zone1. If that gatekeeper fails, the H.323 Service subsequently attempts to register to 10.0.2.100 on port 1719, with the first local zone defined on that gatekeeper.

## Call Disposition

The call dispositions presented in this section apply to both HSRP and alternate gatekeeper.

A gatekeeper can fail in any of the following ways:

- The primary gatekeeper fails
  - Some calls in progress may not be transferred during the period that the endpoints are re-registering to the backup gatekeeper. After the failed transfer, an error is returned to the ICM. If the ICM script is coded to return an error (an END node does this) and survivability is configured on the gateway, the call is default-routed.
  - New calls arriving at the incoming gateway and Unified CVP are serviced correctly, although it is possible that some of the calls might invoke survivability during the period that the endpoints are re-registering to the backup gatekeeper.
• All gatekeepers fail
  – The Unified CVP H.323 Service goes out of service.
  – Calls in progress are not transferred. After the failed transfer, an error is returned to the ICM. If the ICM script is coded to return an error (an END node does this) and survivability is configured on the gateway, the call is default-routed.
  – New calls arriving at the gateway are default-routed if survivability is configured on the gateway.

• The primary gatekeeper degrades but does not fail
  – There are two conditions that usually cause this behavior: low memory due to memory leaks or excessive debug levels causing CPU overload.
  – In this situation, call processing behavior is unpredictable due to the fact that there might be no clean failover to the backup gatekeeper. If survivability is configured on the gateway, calls are default-routed.

**Unified CVP H.323 Service**

When multiple Unified CVP Call Servers are used for redundancy and scalability purposes in Unified CVP, Cisco recommends using a gatekeeper for load balancing and failover services. The H.323 Service is the component of the Unified CVP Call Server that processes H.323 messages and registers with the gatekeeper.

While it is possible for the ingress PSTN gateways to send H.323 calls to the H.323 Service using dial-peers with the specific IP address of the Call Server, doing so results in delays to callers during a failure scenario. In this scenario, a dial-peer is statically configured on the ingress gateways to load-balance across Unified CVP Servers, or in a prioritized fashion so that the primary server is always used under normal conditions. When the H.323 Service is no longer reachable for whatever reason, the dial-peer will attempt to send the call to the failed server and wait for a timeout to occur before proceeding to the next dial-peer configured, and this process occurs for each new call.

When a gatekeeper is used instead, the gateway dial-peer simply points to the gatekeeper, and the gatekeeper is responsible for determining which Call Servers are active and it load balances across them. The gatekeeper's registration process enables it to know which servers are available and does not suffer from the same timeouts as dial-peers. Therefore, Cisco recommends using a gatekeeper instead of static Cisco IOS dial-peers for redundancy and load balancing.

**Configuration**

Unified CVP H.323 configuration for high availability is performed primarily on the ingress gateways, but it is also necessary to configure the H.323 Service to register to the gatekeeper.

**Configuring High Availability for New Calls**

The gatekeeper knows which Unified CVP Call Servers are in service or out of service. It is therefore important to let a gatekeeper route incoming calls to a Unified CVP Call Server. By default, Unified CVP H.323 Services register to the gatekeeper with a technology prefix (tech-prefix) of #2#. The Unified CVP H.323 Service must register with a tech-prefix, and it is not possible to configure the H.323 Service without a tech-prefix.
A technology prefix is a way for the gatekeeper to categorize registering endpoints by functionality. In general, no additional configuration is needed on the gatekeeper for incoming calls. The H.323 Service registers to the gatekeeper with 2#, and the originating gateway prepends a 2# to the incoming Dialed Number Identification Service (DNIS) digits. The gatekeeper automatically knows how to match the gateway request to an available Call Server. On the gatekeeper, the command `show gatekeeper gw-type-prefix` displays the route plan that the gatekeeper uses to route calls.

On the originating gateways, define the dial-peer for the Unified CVP Call Servers as follows:

```
dial-peer voice 11111 voip
  session target ras
  tech-prefix 2#
```

The command `session target ras` instructs the gateway to send the call to its gatekeeper. The command `tech-prefix 2#` instructs the gateway to prepend 2# to the DNIS number when sending the call to the gatekeeper.

### Configuring High Availability for Calls in Progress

In the event that a Unified CVP Call Server fails with calls in progress, it is possible to salvage all calls if certain gateway configuration steps have been taken. A Call Server can fail in one of the following ways:

- The server can crash.
- The process can crash.
- The process can hang.
- There can be a network outage.

The configuration discussed in this section protects against all of these situations. However, the following two situations cannot be protected against:

- Someone stops the process with calls in progress. This situation occurs when a system administrator forgets to put the Call Server out-of-service first to allow calls in progress to finish before stopping the process.
- The Call Server exceeds the recommended call rate. Although there is a throttle for the absolute number of calls allowed in the Call Server, there is no throttle for call rate. In general, exceeding 5 calls per second (cps) for an extended period of time causes the Call Server to have erratic and unpredictable behavior. This situation can be prevented by proper sizing of the Unified CVP system.


The survivability.tcl script itself also contains some directions and useful information.

In the event of most downstream failures (including a Unified CVP Call Server failure), the call is default-routed by the originating gateway. Note that survivability is not applicable in the Unified CVP Standalone and NIC-routing models because there is no Unified CVP Call Server involved anywhere in those models.
Additional Cisco IOS Gateway Configuration

The command in the following example disables the TCP timeout for H.225 signaling on the gateway:

```
voice service voip
h323
no h225 timeout keepalive
```

This action allows the gateway to lose connectivity with the Call Server or Cisco Unified Communications Manager but still retain active calls. If you do not use this command, calls that are still active that are otherwise unaffected by the failure (that is, the RTP stream is still streaming between the endpoints) will be disconnected when the TCP session times out.

The following commands specify the RTP media timeout:

```
ip rtcp report interval 2000
```

When the gateway detects that RTCP messages have not been received in the specified interval, the call is disconnected.

Call Disposition

If the Unified CVP H.323 Service fails, the following conditions apply:

- Calls in progress
  
  If the Unified CVP H.323 Service fails after the caller has been transferred (transfers include transfer to an IP phone, VoiceXML gateway, or other egress gateway), then the call continues normally until a subsequent transfer activity (if applicable) is required from the Unified CVP H.323 Service. If the caller has not hung up and is awaiting further activity, there is a period of 9 to 18 seconds of silence before the call is default-routed by survivability to an alternate location.

  If the call has not yet been transferred, the caller hears 9 to 18 seconds of silence before being default-routed by survivability to an alternate location. (Survivability does not apply in the Unified CVP Standalone and NIC-routing models.)

- New calls
  
  New calls are directed by the gatekeeper to an alternate Unified CVP Call Server. If no Call Servers are available, the call is default-routed to an alternate location by survivability. (Survivability does not apply in Unified CVP Standalone and NIC-routing models.)

Unified CVP IVR Service

With Unified CVP 3.1 and earlier, the IVR Service (previously called the Application Server) was treated independently of the H.323 Service (previously called the Voice Browser) and VoiceXML gateways. High availability was achieved by configuring the Unified CVP Voice Browser and VoiceXML gateways with a list of application server IP addresses and/or using the Content Services Switch (CSS). With Unified CVP 4.0 and later releases, the IVR Service is tightly coupled with the SIP Service or H.323 Service. If the IVR Service goes out of service, the H.323 or SIP Service will be taken out of service as well so that no further calls are accepted by the Call Server.
Configuration

No additional configuration is needed in order to tell the H.323 or SIP Service which IVR Service to use. By default, the H.323 and SIP Service use the IVR Service that resides on the same server. It is also no longer necessary to configure the VoiceXML gateway with the IP address of the Call Server’s IVR Service. When SIP is used, the SIP Service inserts the URL of the Call Server’s IVR Service into a header in the SIP INVITE message when the call is sent to the VoiceXML gateway. The VoiceXML gateway extracts this information from the SIP INVITE and uses it when determining which Call Server to use. When H.323 is used, the VoiceXML gateway examines the source IP address of the incoming call from the Call Server. This IP address is then used as the address for the Call Server’s IVR Service.

The following example illustrates the VoiceXML bootstrap service that is invoked when a call is received:

```
service bootstrap flash:bootstrap.tcl
  paramspace english index 0
  paramspace english language en
  paramspace english location flash
  paramspace english prefix en
```

Unlike Unified CVP 3.1 and earlier releases, with Unified CVP 4.0 and later releases you do not have to configure the IP address of the Call Server. The bootstrap.tcl learns the IP address of the source Call Server and uses it as its call server. There is no need for a CSS or backup Call Server configuration because receiving a call from the Call Server means that the server is up and operational.

The following files in flash memory on the gateway are also involved with high availability: handoff.tcl, survivability.tcl, recovery.vxml, and several .wav files. Use Trivial File Transfer Protocol (TFTP) to load the proper files into flash. Configuration information for each file can be found within the file itself. For more information, refer to the latest version of the *Configuration and Administration Guide for Cisco Unified Customer Voice Portal (CVP)*, available at http://www.cisco.com/en/US/products/sw/custcosw/ps1006/products_installation_and_configuration_guides_list.html

Call Disposition

If the Unified CVP IVR Service fails, the following conditions apply to the call disposition:

- Calls in progress are default-routed to an alternate location by survivability on the originating gateway. (Survivability does not apply in the Unified CVP Standalone and NIC-routing models.)
- New calls are directed to an in-service Unified CVP IVR Service.

VoiceXML Gateway

The VoiceXML gateway parses and renders VoiceXML documents obtained from one or several sources: the Unified CVP Call Server (IVR Service), the Unified CVP VoiceXML Servers, or some other external VoiceXML source. Rendering a VoiceXML document consists of retrieving and playing prerecorded audio files, collecting and processing user input, and/or connecting to an ASR/TTS server for voice recognition and dynamic text-to-speech conversion.
Configuration

High availability configuration for VoiceXML gateways is controlled by the gatekeeper for H.323, the SIP proxy for SIP, and/or the Unified CVP Call Server. Whether the VoiceXML gateways are distributed or centralized also influences how high availability is achieved.

In the event that a Unified CVP Call Server is unable to connect to a VoiceXML gateway, an error is returned to the ICM script. In the ICM script, separate the Send to VRU node from the first Run External script node in order to catch the VoiceXML gateway connection error. If an END script node is used off the X-path of the Send to VRU node, the call is default-routed by survivability on the originating gateway. (Survivability does not apply in the Unified CVP Standalone and VRU-only models.) A Queue to Skill group node could also be used, but that method is effective only if there is an agent available. Otherwise, ICM tries to queue the caller, and that attempt fails because the Unified CVP Call Server is once again unable to connect to a VoiceXML gateway. An END node could then also be used off the X-path of the Queue to Skill Group node to default-route the call.

Centralized VoiceXML Gateways

In this configuration, the VoiceXML gateways reside in the same data center as the Unified CVP Call Server.

H.323 VoiceXML Gateways

On the gatekeeper, configure a zone prefix list that contains the H.323 IDs of all VoiceXML gateways at the data center. For example, assume that there are three VoiceXML gateways in the data center with H.323 IDs of VoiceXMLgw1, VoiceXMLgw2, and VoiceXMLgw3, and that the ICM label for the Network VRU is 5551000. In this example, the gatekeeper distributes calls in essentially a round-robin scheme among all three VoiceXML gateways, provided they are all in service:

```
zone prefix gkzone-name 5551000* gw-priority 10 VoiceXMLgw1 VoiceXMLgw2 VoiceXMLgw3
```

SIP VoiceXML Gateways

If you are using Cisco Unified Presence Server: On the SIP proxy server, configure a static route for the Network VRU label for each gateway. If the VRU label is 5551000, the static route pattern would be 5551000* in order to allow for the correlation-id to be appended and routed to the VoiceXML gateway.

If you are using SIP static routes on the Unified CVP Call Server: Under the SIP Service configuration for the Call Server, configure a static route for each Network VRU label and gateway. If the VRU label is 5551000, the static route pattern would be 5551000>. The > is a wildcard representing one or more digits, and it is needed so that the correlation-id appended to the DNIS number can be passed to the VoiceXML gateway correctly.

In the case of both SIP proxy or Unified CVP static routes, the next-hop address of the route can be either the IP address of the gateway or a DNS SRV record. If you are using an IP address, you must create multiple static routes, one for each VoiceXML gateway. In the case of DNS SRV, only one route per Network VRU label is needed, and the SRV record will provide for load-balancing and redundancy.
Distributed VoiceXML Gateways (Co-Resident Ingress Gateway and VoiceXML)

In this configuration, the gateway that processes the incoming call from the PSTN is separated from the Unified CVP servers by a low-bandwidth connection such as a WAN, and the VoiceXML gateway that is used is the same as the ingress gateway. The purpose of this configuration is to keep the media stream at the edge to avoid consuming bandwidth on the WAN.

H.323 VoiceXML Gateways

Use the SetTransferLabel in VBAdmin (not gatekeeper zone prefixes) to control the selection of the VoiceXML gateway. The SetTransferLabel command is specified per Network VRU label. When the Unified CVP Call Server receives a label from ICM that matches what is configured in the SetTransferLabel, the Unified CVP Call Server performs a gatekeeper lookup but ignores the destination gateway returned by the gatekeeper and sends the call back to the gateway that originated the call. The H.323 Service determines the originating gateway by looking at the source IP address of the H.323 signaling.

SIP VoiceXML Gateways

With SIP, the equivalent of the SetTransferLabel command is the Send to Originator configuration under the SIP Service. If the Network VRU label is 5551000, the Send to Originator pattern would be 5551000>. The > is a wildcard pattern representing one or more digits. The SIP Service determines the originating gateway by looking at the Remote-Party-ID header in the SIP INVITE message.

Distributed VoiceXML Gateways (Separate Ingress Gateway and VoiceXML)

In this configuration, the gateway that processes the incoming call from the PSTN is separated from the Unified CVP servers by a low-bandwidth connection such as a WAN, and the VoiceXML gateway that is used is different than the ingress gateway but located at the same site as the ingress gateway. The purpose of this configuration is to keep the media stream at the same site and not consume bandwidth on the WAN and to optimize VoiceXML gateway sizing when it is appropriate to separate ingress and VoiceXML gateways. In this case, setTransferLabel and Send to Originator cannot be used because you would not want the IVR leg of the call to go back to the ingress gateway. Additionally, it is also impractical to use a gatekeeper or SIP Proxy to control the call routing because you would have to configure separate Network VRUs, Network VRU labels, and customers in ICM for each remote site. Instead, use SetSigDigits functionality.

With this method, the Unified CVP Call Server strips the leading significant digit(s) from the incoming DNIS number. The value that is stripped is saved and prepended when subsequent transfers for the call occur.

H.323 VoiceXML Gateways

When H.323 is used, the significant digit is prepended with a # sign so that the gatekeeper treats it as a technology prefix. The VoiceXML gateway at the remote site should register to the gatekeeper with the same technology prefix as the leading significant digit(s) that were stripped from the DNIS number. The gatekeeper then routes the IVR leg of the call to the correct VoiceXML gateway. If you are using Cisco Unified Communications Manager (Unified CM), remember that Unified CVP indiscriminately prepends the sigdigits value to all transfers, including those to Unified CM. Therefore, when using Unified CM in this scenario, it is necessary to define a gatekeeper-controlled trunk for each of the VoiceXML gateway tech-prefixes and to add zone prefix configuration to the gatekeeper for the Unified CM agents, as illustrated in the following example.
**Configuration of ingress gateway:**

```plaintext
dial-peer voice 1000 voip
tech-prefix 2# (gets the call to CVP)
translate-outgoing called 99
```

Apply a translation-rule to the incoming DNIS number to prepend the value 3:

```plaintext
translation-rule 99
Rule 1 8002324444 38002324444
```

Assuming the DNIS number is 8002324444, the final DNIS string routed to Unified CVP is 2#38002324444.

**Configuration in VB Admin:**

```plaintext
setTechPrefix 2#
setSigDigits 1
```

Strip one digit from the DNIS number after stripping the 2# technology prefix.

**Configuration of VoiceXML gateway:**

Register to the gatekeeper with tech-prefix 3#:

```plaintext
h323-gateway voip tech-prefix 3#
```

**Cisco Unified CM configuration (if used):**

Create a separate gatekeeper-controlled trunk corresponding to each of the tech-prefixes used by the VXML gateways.

**Gatekeeper configuration:**

Define zone prefixes to route calls appropriately to Unified CM agents (only if using Cisco Unified CM).

**Summary of call routing:**

1. A call arrives at Unified CVP with a DNIS string of 2#38002324444.
2. Unified CVP first strips the tech-prefix (2#), leaving 38002324444.
3. Unified CVP then strips one digit (3) from the beginning of the DNIS string, leaving 8002324444.
4. 8002324444 is passed to ICM for call routing.
5. When it is time to transfer, assume ICM returns the label 5551000102. Unified CVP prepends 3#, giving 3#5551000102. This value is then passed to the gatekeeper for address resolution.
6. The gatekeeper resolves this label to the VoiceXML gateway that registered with tech-prefix 3#.
7. The VoiceXML gateway strips the 3#, leaving 5551000102 for the destination address.

**SIP VoiceXML Gateways**

When SIP is used, the significant digits are prepended to the DNIS number, and a SIP Proxy can be configured to route based on those prepended digits. The static routes in the SIP Proxy for the VoiceXML gateway should have the digits prepended. Because these prepended digits were originally populated by the ingress gateway, the SIP Proxy can use them to determine which VoiceXML gateway to use based on the incoming gateway. In this way, calls arriving at a particular site can always be sent back to that site for VoiceXML treatment, with the result that no WAN bandwidth is used to carry the voice RTP stream. Keep in mind that Unified CVP indiscriminately prepends the sigdigits value to all...
transfers, including those to Unified CM. Therefore, when using Unified CM in this scenario, it is necessary to strip the prepended digits when the call arrives so that the real DNIS number of the phone can be used by Unified CM to route the call, as illustrated in the following example.

**Configuration of ingress gateway:**
Apply a translation-rule to the incoming DNIS to prepend the value 3333:

```plaintext
translation-rule 99
  rule 1 8002324444 33338002324444

dial-peer voice 1000 voip
  translate-outgoing called 99
```

Assuming the DNIS number is 8002324444, the final DNIS string routed to Unified CVP is 33338002324444.

**Configuration of Unified CVP SIP Service:**
The Unified CVP SIP Service does not currently have a configuration field for setting the significant digits that should be stripped. Instead, you must edit the sip.properties file. The sip.properties file is located in the C:\Cisco\CVP\conf directory by default. Add the following line to the end of the sip.properties file (to strip four digits from the DNIS number):

```plaintext
SIP.SigDigits = 4
```

**Configuration of VoiceXML gateway:**
Configure the VXML gateway to match the DNIS string, including the prepended digits:

```plaintext
dial-peer voice 3000 voip
  incoming-called number 33335551000T
  service bootstrap
```

Configure the Unified CVP bootstrap.tcl application with the sigdigits parameter, telling it how many digits to strip off of the incoming DNIS string:

```plaintext
application
  service bootstrap flash:bootstrap.tcl
  param sigdigits 4
```

**Cisco Unified CM configuration (if used):**
Configure Unified CM to strip the prepended digits, either by using the Significant Digits configuration on the SIP Trunk configuration page or by using translation patterns.

**SIP Proxy configuration:**
Define static routes on the SIP Proxy, with the prepended digit present, to be sent to the appropriate VoiceXML gateway. Because transfers to agents on a Unified CM cluster will also have the digits prepended, the static routes for agent phones must also contain the prepended digits.

**Summary of call routing:**
1. A call arrives at Unified CVP with a DNIS number of 33338002324444.
2. Unified CVP then strips four digits (3333) from the beginning of the DNIS string, leaving 8002324444.
3. 8002324444 is passed to ICM for call routing.
4. When it is time to transfer, assume ICM returns the label 5551000102. Unified CVP prepends 3333, giving 33335551000102.

5. The SIP Service then resolves the address using the SIP Proxy or local static routes, and it sends the call to the VoiceXML gateway.

6. The VoiceXML gateway bootstrap.tcl will strip the 3333, leaving 5551000102 for the destination address.

**H.323 Alternate Endpoints**

In all cases for either centralized or distributed deployments, configure alternate endpoints for each of the VoiceXML gateways in case the VoiceXML gateway rejects the incoming request (perhaps due to error or overload):

```plaintext
endpoint alt-ep h323id VoiceXMLgw1 ip-address-VoiceXMLgw2
endpoint alt-ep h323id VoiceXMLgw2 ip-address-VoiceXMLgw3
endpoint alt-ep h323id VoiceXMLgw3 ip-address-VoiceXMLgw1
```

**Call Disposition**

If the VoiceXML gateway fails, the following conditions apply to the call disposition:

- Calls in progress are default-routed to an alternate location by survivability on the ingress gateway. (Survivability does not apply in Unified CVP Standalone and NIC-routing models.)
- New calls find an alternate VoiceXML gateway.

**Hardware Configuration for High Availability on the Voice Gateways**

The individual hardware components have the following high-availability options:

- Redundant power supplies and on-hand spares
- Separate components for higher availability
- Dedicated components, which have fewer interaction issues

**Example 1:** Separate PSTN Gateway and VoiceXML Gateway

A PSTN gateway and a separate VoiceXML gateway provide greater availability than a combine PSTN and VoiceXML gateway.

**Example 2:** Duplicate Components for Higher Availability

- Two 8-T1 PSTN gateways provide greater availability than one 16-T1 PSTN gateway.
- Two 96-port VoiceXML servers provide greater availability than one 192-port VoiceXML server.
- Larger designs can use N+1 spares for higher availability.

**Example 3:** Geographic Redundancy for Higher Availability

Geographical redundancy and high availability can be achieved by purchasing duplicate hardware for Side A and Side B.
Content Services Switch (CSS)

The VoiceXML gateway is the only box in the Unified CVP system that makes requests to the CSS. When the VoiceXML gateway needs to make a request for media, ASR/TTS, or VoiceXML, it looks in its configuration to determine to where it should make the request. When a CSS is used, the IP address that is configured on the VoiceXML gateway is a virtual IP address that points to a service configured on the CSS. There are three types of services that the VoiceXML gateway client can request from the CSS:

- Media Server
- ASR/TTS
- Unified CVP VoiceXML Server

If the primary CSS that is servicing these requests should fail, the client VoiceXML gateway must still be able to obtain media and VoiceXML from the servers.

Configuration

You can configure high availability for the CSS by using the Virtual IP (VIP) Redundancy method, as described in the latest version of the Configuration and Administration Guide for Cisco Unified Customer Voice Portal (CVP), available at


Also refer to the latest version of the CSS Redundancy Configuration Guide, available at

http://www.cisco.com/en/US/products/hw/content/ps792/products_installation_and_configuration_guides_list.html

Essentially, a master/backup pair of CSSs functions very much like an HSRP gatekeeper pair. They must reside on the same VLAN and exchange heartbeats using Virtual Router Redundancy Protocol (VRRP), a protocol very similar to HSRP. If the primary CSS fails, the backup CSS recognizes the failure within three seconds and begins processing client requests to its configured virtual IP addresses. The configuration of the master and backup CSSs must always be kept in synchronization.

Call Disposition

If the master CSS fails, then the following conditions apply to the call disposition:

- Calls in progress encounter various behaviors, depending on the type of service the VoiceXML gateway client requested:
  - Media server requests are unaffected.

  The VoiceXML gateway has a very short-lived interaction with the CSS for audio files. Upon receiving a media server request from the gateway, the CSS simply provides an HTTP redirect IP address for the VoiceXML gateway. At that point, the gateway fetches the audio file directly from the media server, bypassing any further interaction with the CSS. Additionally, media file requests to the CSS are very infrequent because the VoiceXML gateway caches previously retrieved media files.
Unified CVP IVR Service requests are unaffected.

Only the initial VoiceXML document request to a Unified CVP IVR Service uses the CSS. The CSS first picks a Unified CVP IVR Service to service the request. The first document passes through the CSS on its return to the VoiceXML gateway. However, subsequent VoiceXML requests are made directly from the VoiceXML gateway client to the Unified CVP IVR Service. If the CSS fails during the very brief period that the first VoiceXML document is being returned, the VoiceXML gateway simply retries the request. If the backup (now primary) CSS selects the same Unified CVP IVR Service as the previous one, there is an error due to a duplicate call instance. In that case, the caller is default-routed by survivability on the originating gateway. (Survivability does not apply in the Unified CVP Standalone model.)

ASR/TTS requests typically fail but might be recoverable.

When the VoiceXML gateway first makes an ASR/TTS request to the CSS, a TCP connection is opened from the VoiceXML gateway to the Media Resource Control Protocol (MRCP) server. That TCP connection goes through the CSS and persists until the caller disconnects or is transferred to an agent. If the primary CSS fails, that TCP connection is terminated. The VoiceXML gateway returns an error code, which you could write a script to work around. The worst-case scenario is that the caller is default-routed to an alternate location by survivability on the originating gateway. (Survivability does not apply in the Unified CVP Standalone model.)

Unified CVP VoiceXML Server requests may fail.

The VoiceXML gateway is "sticky" to a particular Unified CVP VoiceXML Server for the duration of the VoiceXML session. It uses CSS cookies to provide that stickiness. If the CSS fails, the backup CSS has no knowledge of the cookie. Subsequent requests in the session might go to the correct Unified CVP VoiceXML Server, but there is no guarantee. The VoiceXML gateway returns an error code, which you could write a script to work around. The worst-case scenario is that the caller is default-routed to an alternate location by survivability on the originating gateway. (Survivability does not apply in the Unified CVP Standalone model.)

New calls are directed transparently to the VIPs on the backup CSS, and service is unaffected.

Media Server

Audio files can be stored locally in flash memory on the VoiceXML gateway or on an HTTP/TFTP file server. By definition, audio files stored locally are highly available. However, HTTP/TFTP file servers provide the advantage of centralized administration of audio files.

Configuration When Using Unified CVP Microapplications

The VoiceXML gateway sends HTTP requests to an HTTP media server to obtain audio files. It uses the following VoiceXML gateway configuration parameters to locate a server when not using a CSS:

```
ip host mediaserver <ip-address-of-primary-media-server>
ip host mediaserver-backup <ip-address-of-secondary-media-server>
```

The backup server is invoked only if the primary server is not accessible, and this is not a load-balancing mechanism. Once failover occurs, all calls continue to use the backup server until that server becomes inaccessible. Note that mediaserver is not a fixed name, and it needs to match whatever name was assigned to the media_server ECC variable in the ICM script.
The VoiceXML gateway also uses the following VoiceXML gateway configuration parameters to locate a server when using a CSS:

```
  ip host mediaserver <ip-address-of-CSS-VIP-for-media-server>
  ip host mediaserver-backup <ip-address-of-CSS-VIP-for-media-server>
```

Because the CSS almost always locates a media server on the first request, a backup server is rarely invoked. However, it is helpful to configure the backup server when using a CSS for deployments where there are multiple DataCenters with a CSS in each.

**Call Disposition When Using Unified CVP Microapplications**

If the media server fails, the following conditions apply to the call disposition:

- Calls in progress should recover automatically. The high-availability configuration techniques described above should make the failure transparent to the caller. If the media request does fail, use scripting techniques to work around the error (for example, retry the request, transfer to an agent or label, or use TTS).
- New calls are directed transparently to the backup media server, and service is not affected.
- If the media server is located across the WAN from the VoiceXML gateway and the WAN connection fails, the gateway continues to use prompts from gateway cache until the requested prompt becomes stale, at which time the gateway attempts to re-fetch the media and the call fails if survivability is not enabled. If survivability is enabled, the call are default-routed.

**Configuration When Using Unified CVP VoiceXML Studio Scripting**

When scripting in Unified CVP VoiceXML Studio, unlike with ICM scripting, there is no concept of “-backup” for media files. The best the script writer can do is to point `Properties->AudioSettings->Default Audio Path URI` in the application to a single media server or the CSS VIP address for a farm of media servers.

**Unified CVP Video Media Server**

To provide redundancy and load-balancing for the Video Media Server, use a CSS. For configuration details, refer to the CSS configuration and administration guide available at


**Call Disposition**

If the media server fails, the following conditions apply to the call disposition:

- Calls in progress that are streaming from the Video Media Server will stop receiving the media stream. Additionally, the Agent will lose the connection to the Video Media Server and will lose the ability to play media.
- New calls are directed to the backup media server, and service is not affected.
Unified CVP VoiceXML Server

The VoiceXML gateway makes HTTP requests to the Unified CVP VoiceXML Server to obtain VoiceXML documents.

Configuration

The Unified CVP VoiceXML Server high-availability configuration and behavior differ between Standalone deployments and deployments that are integrated with ICM, as described in the following sections.

Standalone Self-Service Deployments


Specifically, it is the CVPPrimaryVXMLServer and CVPBackupVXMLServer gateway parameters that control the high availability characteristics of the VoiceXML server. If VoiceXML server load balancing and more robust failover capabilities are desired, a CSS may be used. (For configuration details, see the latest version of the *Cisco Unified CVP Configuration and Administration Guide.*) Load balancing can also be achieved without a CSS by varying the primary and backup Unified CVP VoiceXML Server configurations across multiple gateways.

Deployments Using ICM

When a VoiceXML Server is used in conjunction with ICM, the ICM script will pass a URL to the VoiceXML gateway in order to invoke the VoiceXML applications. You can configure the ICM script to first attempt to connect to VoiceXML Server A, and if the application fails out the X-path of the VoiceXML Server ICM script node, VoiceXML Server B should be tried. The IP address in the URL can also represent VoiceXML Server VIPs on the CSS.

Call Disposition

If the Unified CVP VoiceXML Server fails, the following conditions apply to the call disposition:

- Calls in progress in a Standalone deployment are disconnected. Calls in progress in an ICM-integrated deployment can be recovered using scripting techniques to work around the error as shown in the script above (for example, retry the request, transfer to an agent or label, or force an error with an END script node to invoke survivability on the originating gateway).
- New calls are directed transparently to an alternate Unified CVP VoiceXML Server. Note that, without a CSS, callers might experience a delay at the beginning of the call and have to wait for the system to timeout while trying to connect to the primary VoiceXML server.
Radvision IVP Server

The Radvision IVP Server communicates with many components of the solution and is central to the redundancy and load-balancing of many of the components of the Full Service Video solution.

The Radvision IVP B2BUA SIP Server receives SIP calls from calling endpoints such as 3G gateways or Cisco Unified Communications Manager servers. A SIP Proxy server should be used to provide redundancy and load-balancing for calls being delivered to multiple Radvision IVP servers.

The Radvision IVP communicates with the Unified CVP Call Servers using an XML interface. The Unified CVP Call Servers register with the Radvision IVP Servers upon initialization, and keepalive messages are exchanged. The Unified CVP Call Servers should be configured to register with all of the Radvision IVP Servers, and the Radvision IVP Servers should be configured to load-balance in a round-robin pattern across all available Unified CVP Call Servers.

The Radvision IVP also communicates with the MCU and EMP modules. The MCU and EMP modules register to the Radvision IVP upon initialization and exchange keepalive messages. Multiple MCU and EMP modules should be pooled together in order to provide redundancy. The Radvision IVP chooses from available MCU and EMP resources as those resources are needed.

Call Disposition

In the event of a Radvision IVP server failure, new calls are delivered by the SIP Proxy to the backup IVP server. Calls in progress are dropped.

In the event of a Unified CVP Call Server failure, registration to the Radvision IVP server is lost and new calls are no longer delivered to the failed Unified CVP Call Server. Calls in progress are dropped. However, if there is a backup Unified CVP Call Server present in the network, then the Radvision IVP server can be configured to use the backup Unified CVP Call Server. This configuration is done via the IVR tab on the backup Unified CVP Call Server.

In the event of an MCU or EMP module failure, the MCU or EMP module will first try to recover itself if it goes offline or if there is an alarm condition. During the recovery period or in a situation where the MCU or EMP is not able to recover itself, it un-registers from the IVP server and is no longer used by IVP for new calls. Calls in progress are dropped.

In the event that the Video Media Server (VMS) is down, no new calls can receive video from agent or video announcements.

Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) Server

The VoiceXML gateway sends MRCP requests to the ASR/TTS servers in order to perform voice recognition and text-to-speech instructions that are defined in a VoiceXML document.

Configuration

The ASR/TTS high-availability configuration and behavior differ between Standalone and ICM-integrated deployments, as described in the following sections.
Standalone Self-Service Deployments

A CSS is required in Standalone deployments to provide failover capabilities for ASR/TTS. For instructions on configuring the CSS for ASR/TTS and on configuring the ASR/TTS Server in a Standalone deployment, see the latest version of the Cisco Unified CVP Configuration and Administration Guide, available at http://www.cisco.com.

Deployments Using ICM

The VoiceXML gateway uses gateway configuration parameters to locate an ASR/TTS server both when using a CSS and when not using a CSS. Note that the backup server is invoked only if the primary server is not accessible and if this is not a load-balancing mechanism. Once failover occurs, all calls continue to use the backup server until that server becomes inaccessible.

The hostnames (such as asr-en-us) are fixed and cannot be modified. The only portion that may be modified is the locale. In the following example, there is a set of primary and backup English ASR/TTS servers and a set of Spanish servers. Configure the CSS, if used, according to the instructions in the latest version of the Configuration and Administration Guide for Cisco Unified Customer Voice Portal (CVP), available at http://www.cisco.com/en/US/products/sw/custcosw/ps1006/products_installation_and_configuration_guides_list.html

When a CSS is used, the IP addresses mentioned the following example would be the virtual IP address for the ASR/TTS service on the CSS.

ip host asr-en-us <ip-address-of-primary-English-ASR-server>
ip host asr-en-us-backup <ip-address-of-secondary-English-ASR-server>
ip host tts-en-us <ip-address-of-primary-English-TTS-server>
ip host tts-en-us-backup <ip-address-of-secondary-English-TTS-server>
ip host asr-es-es <ip-address-of-primary-Spanish-ASR-server>
ip host asr-es-es-backup <ip-address-of-secondary-Spanish-ASR-server>
ip host tts-es-es <ip-address-of-primary-Spanish-TTS-server>
ip host tts-es-es-backup <ip-address-of-secondary-Spanish-TTS-server>

Call Disposition

If the ASR/TTS MRCP server fails, the following conditions apply to the call disposition:

- Calls in progress in Standalone deployments are disconnected. Calls in progress in ICM-integrated deployments can be recovered using scripting techniques to work around the error (for example, retry the request, transfer to an agent or label, switch to prerecorded prompts and DTMF-only input for the remainder of the call, or label or force an error with an END script node to invoke survivability on the originating gateway).

- New calls in Standalone or ICM-integrated deployments are directed transparently to an alternate ASR/TTS server if a CSS is being used. New calls in ICM-integrated deployments are directed transparently to an alternate ASR/TTS server if "-backup" gateway hostnames have been used.
Cisco Unified Communications Manager

Unified CVP transfers callers to Cisco Unified Contact Center Enterprise (Unified CCE) agent phones or desktops using H.323 or SIP. The Unified CVP Call Server receives an agent label from the ICM and routes the call using a gatekeeper or SIP proxy. The call is then sent to the appropriate Cisco Unified Communications Manager (Unified CM) in the cluster, which connects the caller to the agent. The Unified CVP Call Server proxies the call signaling, so it remains in the call signaling path after the transfer is completed. However, the RTP stream flows directly from the originating gateway to the phone. This fact becomes very significant in discussions of high availability.

Configuration

For the most current information on providing Unified CM for high availability, refer to the latest version of the Cisco Unified Contact Center Enterprise Solution Reference Network Design (SRND), available at

http://www.cisco.com/go/srnd

Call Disposition

If the Unified CM process fails on the server that is either hosting the call or hosting the phone, the following conditions apply to the call disposition:

- Calls in progress are preserved. Skinny Client Control Protocol (SCCP) phones have the ability to preserve calls even when they detect the loss of their Unified CM. The caller-and-agent conversation continues until either the caller or agent goes on-hook. The Unified CVP Call Server recognizes that Unified CM has failed, assumes the call should be preserved, and maintains the signaling channel to the originating gateway. In this way, the originating gateway has no knowledge that Unified CM has failed. Note that additional activities in the call (such as hold, transfer, or conference) are not possible. Once the parties go on-hook, the phone then re-homes to another Unified CM server. When the agent goes on-hook, Real-Time Control Protocol (RTCP) packets cease transmitting to the originating gateway, which causes the gateway to disconnect the caller 9 to 18 seconds after the agent goes on-hook. If survivability has been configured on the gateway and the caller is waiting for some additional activity (the agent might think the caller is being blind-transferred to another destination), the caller is default-routed to an alternate location.

- New calls are directed to an alternate Unified CM server in the cluster.
Intelligent Contact Management (ICM)

Cisco Intelligent Contact Management (ICM) software provides enterprise-wide distribution of multichannel contacts (inbound/outbound telephone calls, Web collaboration requests, email messages, and chat requests) across geographically separated contact centers. ICM software is an open standards-based solution whose capabilities include routing, queuing, monitoring, and fault tolerance.

Configuration

For the most current information on configuring ICM for high availability, refer to the latest version of the Cisco Unified Contact Center Enterprise Solution Reference Network Design (SRND), available at http://www.cisco.com/go/srnd

Call Disposition

There are many components in Cisco ICM, and call disposition varies depending on the component that fails. Although there are a few exceptions, the following conditions apply to the call disposition:

- If the Voice Response Unit (VRU) Peripheral Gateway (PG) or any component on the VRU PG fails, calls in progress are default-routed by survivability on the originating gateway.
- If the Logger fails, calls in progress are unaffected.
- If the primary router fails, calls in progress are unaffected. If both the Side A and Side B routers fail, calls in progress are default-routed by survivability on the originating gateway.
- New calls are directed to the backup ICM component.
Interactions with Cisco Unified ICM

This chapter discusses Cisco Unified Intelligent Contact Management (ICM) from the perspective of its relationship with Unified CVP. In some cases, the choice of deployment model has implications for Unified ICM; and in other cases, certain choices about the Unified ICM configuration carry implications for the Unified CVP deployment.

This chapter covers the following topics:

- Network VRU Types, page 5-1
- Network VRU Types and Unified CVP Deployment Models, page 5-5
- Hosted Implementations, page 5-9
- Deployment Models and Sizing Implications for Calls Originated by Cisco Unified Communications Manager and ACDs, page 5-13
- Using Third-Party VRUs, page 5-14

Network VRU Types

This section first discusses Network VRU types for Unified ICM in general, then it discusses them as they relate to Unified CVP deployments in particular.

This section covers the following topics:

- Overview of Unified ICM Network VRUs, page 5-2
- Unified CVP as a Type 10 VRU, page 5-3
- Unified CVP as Type 5 VRU, page 5-3
- Unified CVP as Type 3 or 7 VRU (Correlation ID Mechanism), page 5-4
- Unified CVP as Type 8 or 2 VRU (Translation Route ID Mechanism), page 5-5

In this document, the terms voice response unit (VRU) and interactive voice response (IVR) are used interchangeably.
Overview of Unified ICM Network VRUs

This section describes the types of Unified ICM VRUs used for Unified CVP applications. Unified ICM perceives calls that need IVR treatment as having two portions: the Switch leg and the VRU leg. The Switch is the entity that first receives the call from the network or caller. The VRU is the entity that plays audio and performs prompt-and-collect functions. Unified CVP can participate in the Switch role or the VRU role, or both, from the perspective of Unified ICM. In a network deployment, multiple Unified CVP devices can also be deployed to provide the Switch and VRU portions independently.

The call delivery to VRU can be based on either a Correlation ID or a translation route mechanism, depending on the network capability to pass the call reference identification to the VRU. Call reference identification is needed because Unified ICM has to correlate the two legs of the same call in order to provide instructions for completing the call. In the Unified ICM application, the VRU has to supply this call reference ID to Unified ICM when the VRU asks for instructions on how to process the incoming call that it receives from the switch. This mechanism enables Unified ICM to retrieve the appropriate call context for this same call, which at this stage is to proceed to the IVR portion of the call. These two correlation mechanisms operate as follows:

- **Correlation ID**
  This mechanism is used if the network can pass the call reference ID to the VRU, which is usually the case when the VRU is located in the network with the switch and the call signaling can carry this information (for example, the Correlation ID information is appended to the dialed digits when Unified ICM is used). This mechanism usually applies to calls being transferred within the VoIP network.

- **Translation Route ID**
  This mechanism is used when the VRU is reachable across the PSTN (for example, the VRU is at the customer premise) and the network cannot carry the call reference ID information in delivering the call to the VRU. A temporary directory number (known as a translation route label) has to be configured in Unified ICM to reach the VRU, and the network routes the call normally to the VRU as with other directory number routing in the PSTN. When the VRU asks for instructions from Unified ICM, the VRU supplies this label (which could be a subset of the received digits) and Unified ICM can correlate the two portions of the same call. Normally the PSTN carrier will provision a set of translation route labels to be used for this purpose.

---

**Note**

The deployed VRU can be located in the network (Network VRU) or at the customer premises. In the latter scenario, a Network Applications Manager (NAM) would be deployed in the network and a Customer ICM (CICM) would be deployed at the customer premises. The corresponding Correlation ID or Translation Route ID should be used accordingly, as described earlier, depending on the location of the VRU.
Unified CVP as a Type 10 VRU

Type 10 was designed to simplify the configuration requirements in Unified CVP Comprehensive Model deployments. The Type 10 VRU is the preferred VRU Type for all new installations, but it requires Cisco Unified ICM 7.1. Unified ICM 7.0 deployments should use the VRU types outlined in subsequent sections of this chapter. The Type 10 VRU does not support ICM Customers, which is an ICM feature that allows you to have multiple Network VRU’s and is typically used in Hosted deployments. Deployments that need to use ICM Customers can use the other VRU types outlined below.

Type 10 Network VRU has the following behavior:

- There is a Handoff of routing client responsibilities to the Unified CVP switch leg.
- There is an automatic transfer to the Unified CVP VRU leg, resulting in a second transfer in the case of calls originated by the VRU, ACD, or Cisco Unified Communications Manager (Unified CM).
- For calls originated by Unified CM, the Correlation ID transfer mechanism is used. The Correlation ID is automatically added to the end of the transfer label defined in the Type 10 Network VRU configuration.
- The final transfer to the Unified CVP VRU leg is similar to a Type 7 transfer, in that a RELEASE message is sent to the VRU prior to any transfer.

In Unified CVP implementations, a single Type 10 Network VRU should be defined, and all Unified ICM VRU scripts should be associated with it. It requires one label for the Unified CVP Switch leg routing client, which will transfer the call to the Unified CVP VRU leg. If calls will be transferred to Unified CVP from Unified CM, it also needs another label for the Unified CM routing client, and this label should be different from the label used for the CVP Routing Client. This label will transfer the call to the Unified CVP Switch leg. The Unified ICM Router will send this label to Unified CM with a Correlation ID concatenated to it. Unified CM must be configured to handle these arbitrary extra digits.

The Unified CVP Switch leg peripheral should be configured to point to the same Type 10 Network VRU. Also, all incoming dialed numbers for calls that are to be transferred to Unified CVP should be associated with a Customer Instance that points to the same Type 10 Network VRU.

For calls that originate at a Call Routing Interface VRU or at a TDM ACD, a TranslationRouteToVRU node should be used to transfer the call to Unified CVP’s Switch leg peripheral. For all other calls, use either a SendToVRU node, a node that contains automatic SendToVRU behavior (such as the queuing nodes), or a RunExternalScript.

Unified CVP as Type 5 VRU

Note

Cisco Unified ICM 7.1 introduces the Type10 Network VRU. This VRU should be used for all new implementations of Unified CVP using Unified ICM 7.1 or greater. The Type5 VRU can still be used for existing customer deployments that have upgraded or for deployments that are not running Unified ICM 7.1 or later.

Type 5 and Type 6 are similar in the sense that the VRU entity functions both as a switch (call control) and as the VRU (IVR). However, they differ on how to connect to the VRU.

In Type 6, the Switch and the VRU are the same device, therefore the call is already at the VRU. No Connect and Request Instructions message sequence is needed from the point of view of Unified ICM.
On the other hand, in Type 5, the Switch and the VRU are different devices even though they are in the same service node from the viewpoint of Unified ICM (that is, they both interact with Unified ICM through the same PG interface). Therefore, Unified ICM uses a Connect and Request Instructions sequence to complete the IVR call.

**Note**

In a Unified CVP application, there are two legs of the call as perceived by Unified ICM: the Switch leg and the VRU leg. In the case where Unified CVP acts as the service node application (that is, when Unified CVP receives the call from the network directly and not via pre-routing), Unified CVP will appear to Unified ICM as Type 5 because the call control (Unified CVP) and the VRU device are different. Hence, Unified CVP must be configured as VRU Type 5 in the Unified ICM and NAM configuration for the Switch leg. The VRU leg requires a different setup, depending on the deployment model (for example, the VRU leg could be Type 7 in the Comprehensive Unified ICM enterprise deployment model). For examples of configuring Unified CVP as Type 5, refer to the latest version of the *Configuration and Administration Guide for Cisco Unified Customer Voice Portal (CVP)*, available at [http://www.cisco.com/en/US/products/sw/custcosw/ps1006/products_installation_and_configuration_guides_list.html](http://www.cisco.com/en/US/products/sw/custcosw/ps1006/products_installation_and_configuration_guides_list.html).

Neither Correlation ID nor Translation Route ID is needed when Unified CVP acts as a Type 5 VRU to Unified ICM and the NAM. However, a dummy label is sometimes required.

### Unified CVP as Type 3 or 7 VRU (Correlation ID Mechanism)

**Note**

Cisco Unified ICM 7.1 introduces the Type10 Network VRU. This VRU should be used for all new implementations of Unified CVP using Unified ICM 7.1 or greater, except as VRU Only (Model #4a, described below). The Type 3 or 7 VRU can still be used for existing customer deployments that have upgraded or for deployments that are not running Unified ICM 7.1 or later.

When the VRU functions as an IVR with the Correlation ID mechanism, Unified ICM uses Type 3 and Type 7 to designate sub-behaviors of the VRU via the PG in the Correlation ID scheme. Both Type 3 and Type 7 VRUs can be reached via the Correlation ID mechanism, and a PG is needed to control the VRU. However, the difference between these two types is in how they release the VRU leg and how they connect the call to the final destination.

In Type 3, the switch that delivers the call to the VRU can take the call from the VRU and connect it to a destination (or agent).

In Type 7, the switch cannot take the call away from the VRU. When the IVR treatment is complete, Unified ICM must disconnect or release the VRU leg before the final connect message can be sent to the Switch leg to instruct the switch to connect the call to the destination.

When used as an Intelligent Peripheral IVR, Unified CVP can function with either Type 3 or 7, but it is somewhat more efficient under Type 7 because it gets a positive indication from Unified ICM when its VRU leg is no longer needed (as opposed to waiting for the VoiceXML gateway to inform it that the call has been pulled away).

As stated previously, there are two legs of the call: the Switch leg and the VRU leg. Different Unified CVP hardware can be used for each leg, but from the perspective of Unified ICM functionality, there will be a Unified CVP via PG acting as VRU Type 5 (that is, a service node) along with potentially a different Unified CVP via another PG acting as VRU Type 7 to complete the IVR application (self service, queuing, and so forth).
For configuration examples of the Unified CVP application with VRU Type 3 or Type 7, refer to the latest version of the Configuration and Administration Guide for Cisco Unified Customer Voice Portal (CVP), available at


**Unified CVP as Type 8 or 2 VRU (Translation Route ID Mechanism)**

Cisco Unified ICM 7.1 introduces the Type10 Network VRU. This VRU should be used for all new implementations of Unified CVP using Unified ICM 7.1 or greater, except as VRU Only (Model #4a, described below). The Type 8 or 2 VRU can still be used for existing customer deployments that have upgraded or for deployments that are not running Unified ICM 7.1 or later.

When the VRU functions as an IVR with the Translation Route ID mechanism, Unified ICM uses Type 8 or Type 2 to designate sub-behaviors of the VRU via the PG in the translation route scheme. Both Type 2 and Type 8 VRUs can be reached via the Translation Route mechanism, and PG is needed to control the VRU. However, they differ in how they connect the call to the final destination.

In Type 8, the switch that delivers the call to the VRU can take the call from the VRU and connect it to a destination/agent.

Type 2 is used when the switch does not have the ability to take the call away from the VRU to deliver it to an agent. In that case, when the IVR treatment is complete, Unified ICM sends the final connect message to the VRU (rather than to the original switch) to connect the call to the destination. The VRU effectively assumes control of the switching responsibilities when it receives the call. This process is known as a handoff.

Similarly to the Correlation ID case, there are two legs of the call: the Switch leg and the VRU leg. Unified CVP can be used for either the Switch leg or the VRU leg. For example, when a Network Interface Controller (NIC), NAM, or CICM is involved, Unified CVP should be configured as Type 2 or Type 8 in the VRU leg.

For configuration examples of the Unified CVP application with VRU Type 8 or Type 2, refer to the latest version of the Configuration and Administration Guide for Cisco Unified Customer Voice Portal (CVP), available at


**Network VRU Types and Unified CVP Deployment Models**

This section describes how Network VRU types relate to the Unified CVP deployment models described in the chapter on Functional Deployment Models, page 2-1. This section covers the following topics:

- **Model #1: Standalone Self-Service**, page 5-7
- **Model #2: Call Director**, page 5-7
- **Model #3a: Comprehensive Using ICM Micro-Apps**, page 5-7
- **Model #3b: Comprehensive Using CVP VoiceXML Server**, page 5-7
In Unified ICM, a Network VRU is a configuration database entity. It is accessed using the Network VRU Explorer. A Network VRU entry contains the following pieces of information:

- **Type** — A number from 2 to 10, which corresponds to one of the types described previously.
- **Labels** — A list of labels that Unified ICM can use to transfer a call to the particular Network VRU being configured. These labels are relevant only for Network VRUs of Type 3, 7, or 10 (that is, those VRU types that use the Correlation ID mechanism to transfer calls), and they are required but never used in the case of Type 5. Each label consists of two parts:
  - A digit string, which becomes a DNIS that can be understood by the gatekeeper (when H.323 is used), by a SIP Proxy Server or a static route table (when SIP is used), or by gateway dial peers.
  - A routing client, or switch leg peripheral. In other words, each peripheral device that can act as a Switch leg must have its own label, even though the digit strings will likely be the same in all cases.

Network VRU configuration entries themselves have no value until they are associated with active calls. There are three places in Unified ICM where this association is made:

- Under the Advanced tab for a given peripheral in the PG Explorer tool
- In the Customer Instance configuration in the Unified ICM Instance Explorer tool
- In every VRU Script configuration in the VRU Script List tool

Depending on the protocol-level call flow, Unified ICM Enterprise looks at either the peripheral or the Customer Instance to determine how to transfer a call to a VRU. Generally speaking, Unified ICM Enterprise examines the Network VRU that is associated with the Switch leg peripheral when the call first arrives on a Switch leg, and the Network VRU that is associated with the VRU leg peripheral when the call is being transferred to the VRU using the Translation Route mechanism. It examines the Network VRU that is associated with the Customer Instance when the call is being transferred to the VRU using the Correlation ID mechanism.

Unified ICM Enterprise also examines the Network VRU that is associated with the VRU Script every time it encounters a RunExternalScript node in its routing script. If Unified ICM does not believe the call is currently connected to the designated Network VRU, it will not execute the VRU Script.

Unified ICM Enterprise Release 7.1 introduced Network VRU Type 10, which simplifies the configuration of Network VRUs for Unified CVP. For most call flow models, a single Type 10 Network VRU can take the place of the Type 2, 3, 7, or 8 Network VRUs that were associated with the Customer Instance and/or the switch and VRU leg peripherals. The only major call flow model that still requires Type 7 or 8 is VRU Only (Model #4a, described below).

Note that the previously recommended VRU types still work as before, even in Unified ICM Enterprise 7.1. New installations should use Type 10 if possible, and existing installations may optionally switch to Type 10.
Model #1: Standalone Self-Service

The Standalone Self-Service model typically does not interface with Unified ICM VRU scripts, so a Network VRU setting is not relevant. The Standalone Self-Service model with Unified ICM Label Lookup does not use the VRU scripts in Unified ICM; it simply issues a Route Request to the VRU PG Routing Client, therefore a Network VRU is not needed.

Model #2: Call Director

In this model, Unified ICM (and therefore Unified CVP) is responsible for call switching only. It does not provide queuing or self-service, so there is no VRU leg. Therefore, a Network VRU setting is not required in this case.

Model #3a: Comprehensive Using ICM Micro-Apps

In this model, Unified CVP devices act as both the Switch and the VRU leg, but the call does need to be transferred from the Switch leg to the VRU leg before any call treatment (playing .wav files or accepting user input) can take place. Associate all Unified CVP peripherals with a Type 10 Network VRU in this case.

Note

Type10 is available only in Unified ICM 7.1 and later, and new implementations should use this configuration. For Unified ICM 7.0, a Type 2 Network VRU should be used in this case.

Associate all incoming dialed numbers with a Customer Instance that is associated with a Type 10 Network VRU. All the VRU Scripts that will be executed by this call must be associated with the same Type 10 Network VRU. Although it is not always necessary, the best practice is for the Unified ICM routing script to execute a SendToVRU node prior to the first RunExternalScript node.

Note

Type10 is available only in Unified ICM 7.1 and later, and new implementations should use this configuration. For Unified ICM 7.0, a Type 7 should be used in this case.

Model #3b: Comprehensive Using CVP VoiceXML Server

From the perspective of call routing and the Network VRU, this model is identical to Model #3a, described above.
Model #4: VRU Only

In this model, the call first arrives at Unified ICM through an ICM-NIC interface, not through Unified CVP. At least initially, Unified CVP is not responsible for the Switch leg; its only purpose is as a VRU. However, depending on which kind of NIC is used, it might be required to take over the Switch leg once it receives the call. This model actually has two submodels, which we are described separately in the following sections.

Model #4a. VRU Only with NIC Controlled Routing

This submodel assumes a fully functional NIC that is capable of delivering the call temporarily to a Network VRU (that is, to Unified CVP's VRU leg) and then retrieving the call and delivering it to an agent when that agent is available. It further assumes that, if the agent is capable of requesting that the call be re-transferred to another agent or back into queue or self-service, the NIC is capable of retrieving the call from the agent and re-delivering it as requested.

There are two variants of this submodel, depending on whether the Correlation ID or the Translation Route mechanism is used to transfer calls to the VRU. Most NICs (actually, most PSTN networks) are not capable of transferring a call to a particular destination directory number and carrying an arbitrary Correlation ID along with it, which the destination device can pass back to Unified ICM in order to make the Correlation ID transfer mechanism function properly. For most NICs, therefore, the Translation Route mechanism must be used.

There are a few exceptions to this rule, however, in which case the Correlation ID mechanism can be used. The NICs that are capable of transmitting a Correlation ID include Call Routing Service Protocol (CRSP), SS7 Intelligent Network (SS7IN), and Telecom Italia Mobile (TIM). However, because this capability also depends on the PSTN devices that connect behind the NIC, check with your PSTN carrier to determine whether the Correlation ID can be passed through to the destination.

If the NIC is capable of transmitting the Correlation ID, the incoming dialed numbers must all be associated with a Customer Instance that is associated with a Type 7 Network VRU. The Type 7 Network VRU must contain labels that are associated to the NIC routing client, and all the VRU Scripts must also be associated with that same Type 7 Network VRU. The peripherals need not be associated with any Network VRU. Although it is not always necessary, the best practice is for the Unified ICM routing script to execute a SendToVRU node prior to the first RunExternalScript node.

If the NIC is not capable of transmitting a Correlation ID (the usual and safe case), then the incoming dialed numbers must all be associated with a Customer Instance that is not associated with any Network VRU. The Unified CVP peripherals must, however, be associated with a Network VRU of Type 8, and all the VRU Scripts must also be associated with that same Type 8 Network VRU. In this case it is always necessary to insert a TranslationRouteToVRU node in the routing script prior to the first RunExternalScript node. If the call is going to the VRU leg because it is being queued, generally the TranslationRouteToVRU node should appear after the Queue node. In that way, an unnecessary delivery and removal from Unified CVP can be avoided when the requested agent is already available.

Model #4b. VRU Only with NIC Controlled Pre-Routing

This submodel assumes a less capable NIC that can deliver the call only once, whether to a VRU or to an agent. Once the call is delivered, the NIC cannot be instructed to retrieve the call and re-deliver it somewhere else. In these cases, Unified CVP can take control of the switching responsibilities for the call. From the perspective of Unified ICM, this process is known as a handoff.

Calls that fit this particular submodel must use the Translation Route mechanism to transfer calls to the VRU. There is no way to implement a handoff using the Correlation ID mechanism.
To implement this model with Unified ICM 7.1, the incoming dialed numbers must all be associated with a Customer Instance that is associated with a Type 10 Network VRU. The VRU labels are associated with the Unified CVP routing client, not the NIC. The Unified CVP peripherals and VRU Scripts must be associated with the Type 10 Network VRU. In this case, it is always necessary to insert a TranslationRouteToVRU node in the routing script, followed by a SendToVRU node, prior to the first RunExternalScript node. If the call is going to the VRU leg because it is being queued, generally these two nodes should appear after the Queue node. In that way, an unnecessary delivery and removal from Unified CVP can be avoided if the requested agent is already available.

To implement this model with Unified ICM 7.0, the incoming dialed numbers must all be associated with a Customer Instance that is associated with a Type 7 Network VRU. The VRU labels are associated with the Unified CVP routing client, not the NIC. The Unified CVP peripherals must be associated with a Network VRU of Type 2, but all the VRU Scripts must be associated with the Type 7 Network VRU. In this case, it is always necessary to insert a TranslationRouteToVRU node in the routing script, followed by a SendToVRU node, prior to the first RunExternalScript node. If the call is going to the VRU leg because it is being queued, generally these two nodes should appear after the Queue node. In that way, an unnecessary delivery and removal from Unified CVP can be avoided if the requested agent is already available.

Note

Two different VRU transfer nodes are required. The first one transfers the call away from the NIC with a handoff, and it establishes Unified CVP as a Switch leg device for this call. Physically the call is delivered to an ingress gateway. The second transfer delivers the call to the VoiceXML gateway and establishes Unified CVP as the call's VRU device as well.

Hosted Implementations

This section covers the following topics:

- Overview of Hosted Implementations, page 5-9
- Using Unified CVP in Hosted Environments, page 5-10
- Unified CVP Placement and Call Routing in a Hosted Environment, page 5-11
- Network VRU Type in a Hosted Environment, page 5-12

Overview of Hosted Implementations

Hosted implementations incorporate a two-level hierarchy of Unified ICM systems. The Network Application Manager (NAM) sits at the top level, and one or more Customer ICMs (CICMs) sit below it. Both the NAM and CICM are really complete ICMs in and of themselves, with a communication link between them known as Intelligent Network Call Routing Protocol (INCRP). Each CICM acts in an isolated fashion; it is not aware of the other CICMs, nor is it aware that the NAM is itself another ICM. It has no connection to the other CICMs, but its connection to the NAM is through a NIC – specifically, the INCRP NIC.

Traditionally, customers implement Hosted setups because they are service providers. They want to provide ICM contact center services to multiple customers of their own. Each customer is hosted on its own CICM, and the NAM is responsible for routing calls, which are delivered to the service provider, to the appropriate customer's CICM. The individual customers run their own contact centers with their own ACDs connected to PGs at their own premises. The PGs, in turn, are connected to their assigned CICMs at the service provider. Thus, the service provider owns and hosts the NAM and all the CICMs,
but all the ACDs are owned and hosted by the individual customers. The PGs for those ACDs are owned by the service provider but are located at the customer's premises, next to the ACDs. The service provider itself does not necessarily operate any ACDs of its own, but it could; those PGs could be connected to a CICM that is assigned to the service provider, or they could actually be connected to the NAM itself.

In terms of ICM scripting, all incoming calls initially invoke an appropriate NAM routing script that has the primary responsibility of identifying the appropriate target customer. The script then delegates control to a routing script that is running on that customer's CICM. The CICM-based routing script can then select the appropriate ACD to which to deliver the call, and it can return the necessary translation route label to the NAM. The NAM can then instruct its routing client to deliver the call to the designated target ACD. If the CICM routing script determines that no ACD can currently take the call or that it cannot yet identify which ACD should take the call, it can ask the NAM to place the call into queue at a Service Control VRU. The CICM routing script can then issue Network VRU Script requests via the NAM to that VRU until a routing decision is made.

In practice, however, the NAM and CICM architecture is flexible enough to enable a number of other possibilities. Many Hosted customers use this topology simply as a way to get more calls or more PGs through their ICM setup. Others use CICMs, not for customer contact centers, but for outsourcers. In such cases, the NAM handles perhaps the same number of calls as the CICM, and the CICM machines themselves might be located quite far away from the NAM. Also, the NAM and CICM architecture was designed at a time when all contact centers ran on TDM-based ACDs. The addition of VoIP routing and ACDs based on Unified CM (that is, Unified CCE) with direct agent routing made matters considerably more complicated.

**Using Unified CVP in Hosted Environments**

When Unified CVP is involved, it is usually used as a self-service and/or queuing platform connected to the NAM and physically located within the service provider's data center. Thus, it enables the traditional service provider not only to route calls to the appropriate customer-owned ACDs but also to retain control of calls that are queued for those ACDs and to provide either basic prompt-and-collect capability or full-featured self-service applications to its customers. The latter case typically incorporates Unified CVP VoiceXML Servers into the network. Depending on the customer's needs, the service provider might host the VoiceXML Servers or the customer might host them. Additionally, the service provider might write and own the self-service application, or the customer might write and own them. Allowing the customer to own or host the VoiceXML Servers is a convenient solution when the self-service application needs to reference back-end services. It allows the customer to keep control of that interaction within its own enterprise network, while transmitting only VoiceXML over HTTP to the service provider's VoiceXML gateway.

In many Hosted environments, particularly when the service provider is itself a PSTN carrier, all the actual call routing occurs via an ICM NIC. In that sense, these deployments are very much like Deployment Model #4, VRU Only with NIC Controlled Routing. The same situation applies if a PGW is being used to route calls using (typically) the ICM SS7 NIC. However, quite often the service provider does not have a NIC interface at all, and all calls arrive via TDM interfaces such as T3 or E3. In those cases, Unified CVP is used as the Switch leg as well as the VRU leg. This situation is similar to Model #3a, Comprehensive Using ICM Micro-Apps, or to Model #3b, Comprehensive Using CVP VoiceXML Server.
Unified CVP Placement and Call Routing in a Hosted Environment

As described previously, if Unified CVP is used in its traditional way as a true Network VRU, it is usually connected at the NAM. However, various requirements might cause Unified CVP to be placed at the CICM level instead, or in addition. The following guidelines apply when considering where to place Unified CVP components:

- If Unified CVP is placed at the NAM and Unified CVP handles both the Switch leg and the VRU leg, use the Correlation ID transfer mechanism. The SendToVRU node may be executed by either the NAM or the CICM routing script. (The RunExternalScript nodes should also be in the same script that executed the SendToVRU.)

- If Unified CVP is placed at the NAM and a NIC handles the Switch leg while Unified CVP handles the VRU leg, either the Correlation ID transfer mechanism or the Translation Route transfer mechanism may be used, depending on the capabilities of the NIC. (See Model #4a. VRU Only with NIC Controlled Routing, page 5-8.) In this case, the following guidelines also apply:
  - If Correlation ID transfers are used, then the SendToVRU node may be contained in either the NAM or the CICM routing script. (The RunExternalScript nodes should also be in the same script that executed the SendToVRU.)
  - If Translation Route transfers are used, then the TranslationRouteToVRU node, together with all RunExternalScript nodes, must be in the NAM routing script. The implication here is that the call is queued (or treated with prompt-and-collect) before the particular CICM is selected. This configuration does not make much sense for queuing, but it could be useful for service providers who want to offer initial prompt-and-collect before delegating control to the CICM.

- If Unified CVP is placed at the CICM and a NIC handles the Switch leg while Unified CVP handles the VRU leg, only the Translation Route transfer method can be used. The TranslationRouteToVRU node, together with all RunExternalScript nodes, must be in the CICM routing script.

Adding calls initiated by Unified CM or an ACD brings additional constraints. Both of these devices are considered ACDs from the ICM’s perspective, and they most likely are connected at the CICM level. Assuming these are new calls (as opposed to continuations of existing calls), the route request will come from the ACD and the resulting label will be returned to the ACD. Neither Unified CM nor any ACD is capable of transmitting a Correlation ID upon transfer; only the Translation Route transfer method can be used. This limitation further implies that the transfer destination (for example, Unified CVP) must also be connected at the CICM level, not the NAM level.

If the calls are not new but continuations of existing calls, then they are attempts to transfer an existing inbound caller from one agent to another agent. Customers might want these transfers to be either blind network transfers (that is, the first agent drops off and the caller is delivered to a second agent or queued for a second agent) or warm consultative transfers (that is, the caller goes on hold while the first agent speaks to a second agent or waits in queue for a second agent and eventually hangs up, leaving the caller talking to the second agent). The following guidelines apply to such transfers:

- **Blind network transfers**
  - Whether or not the original call was introduced to the NAM via a NIC or Unified CVP Switch leg, the transfer label will be passed from the CICM to the NAM to the original Switch leg device. There are two sub-cases of blind network transfers:
    - If the Switch leg device is Unified CVP or a NIC that can handle Correlation ID, the Correlation ID transfer mechanism can be used. The SendToVRU node and all RunExternalScript nodes must be incorporated in the CICM routing script. The Unified CVP VRU leg can be connected to the NAM. This combination of blind transfer with Correlation ID transfer is ideal for Unified CVP and should be employed if at all possible.
If the Switch leg device is a NIC that cannot handle Correlation ID, then the Translation Route transfer method must be used, which further implies that the Unified CVP VRU leg device must be connected to the CICM.

**Note**
In this case, the customer might have to deploy additional dedicated Unified CVP Call Servers at the CICM level because the NAM-level Unified CVP Call Servers cannot be used.

- Warm consultative transfers

Unified CVP provides warm consultative transfers only in the case of Unified CCE agents transferring calls to other Unified CCE agents, where Unified CVP owns the initial Switch leg for the inbound call. For TDM agents, the ACD's own mechanisms are used and Unified CVP is not involved. In the case where the incoming calls to Unified CCE agents arrive through a NIC, Unified ICM's Network Consultative Transfer facility can be used, and it also would not involve Unified CVP.

In the one supported case where Unified CVP owns the initial Switch leg and the transfer is among Unified CCE agents, the Translation Route transfer method must be used because Unified CM cannot handle Correlation ID transfers. Again, this means that the Unified CVP VRU leg device must be connected to the CICM.

**Note**
In this case, the customer might have to deploy additional dedicated Unified CVP Call Servers at the CICM level because the NAM-level Unified CVP Call Servers cannot be used.

### Network VRU Type in a Hosted Environment

In Hosted environments, there are always two ICM systems to consider: the NAM and the CICM in question. Network VRU Types are configured differently in the NAM than in the CICM.

The NAM, as described earlier, sees new calls arrive either from a NIC or from Unified CVP, and it is aware of the Unified CVP VRU leg device. The NAM's Network VRU Types must be configured exactly as if it were an independent ICM operating with those devices. The fact that transfer labels sometimes come from a CICM can be ignored for the purposes of configuring Network VRU Types.

The CICM, on the other hand, sees new calls arrive from a NIC – the Intelligent Network Call Routing Protocol (INCRP) NIC, to be specific. All the dialed numbers that can arrive from the NAM must be associated with a Customer Instance that is associated with a Type 7 Network VRU. Associate that Network VRU with all VRU Scripts, and provide the same label as you need in the NAM's Network VRU definition, but with the INCRP NIC as its routing client. Other than that, no peripherals have Network VRUs configured.
Deployment Models and Sizing Implications for Calls Originated by Cisco Unified Communications Manager and ACDs

The information in this section applies to all ACDs as well as to all Cisco Unified Communications Manager (Unified CM) integrations that use Unified CVP rather than Cisco IP IVR for queuing. As far as Unified CVP is concerned, these devices share the following characteristics:

- They manage agents and can therefore be destinations for transfers.
- They can issue Route Requests and can therefore be Switch leg devices.
- Although they can be Switch leg devices, they cannot handle more than one transfer and they might not be able to handle the Correlation ID.

A Unified CM or ACD user would typically issue a Route Request for one of the following reasons:

- To be connected to another agent in a particular skill group
- To reach a self-service application
- To blind-transfer a previously received call to one of the above entities

In addition, a Unified CM user in particular might issue a Route Request for one of the following reasons:

- To deliver a successful outbound call from the Unified ICM Outbound dialer to a self-service application based on Unified CVP
- To warm-transfer a call that the user had previously received to either a particular skill group or a self-service application

Each of the above calls invokes an Unified ICM routing script. The script might or might not search for an available destination agent or service. If it finds an appropriate destination, it sends the corresponding label either back to the ACD or, if blind-transferring an existing call, to the original caller's Switch leg device. If it needs to queue the call or if the ultimate destination is intended to be a self-service application rather than an agent or service, the script sends a VRU translation route label either back to the ACD or, if blind-transferring an existing call, to the original caller's Switch leg device.

If the above sequence results in transferring the call to Unified CVP's VRU leg device, there is a second transfer to deliver it to a VoiceXML gateway. To ensure that these events take place, the following Unified ICM configuration elements are required:

- For new calls from the ACD or warm transfers of existing calls:
  - The Unified CVP peripheral must be configured to be associated with a Type 10 Network VRU (Type 2 if Unified ICM 7.0 is used).
  - The dialed number that the ACD dialed must be associated with a Customer Instance that is associated with a Type 10 Network VRU (Type 7 if Unified ICM 7.0 is used).
  - With Unified ICM 7.0, or with a Unified ICM 7.1 and an ACD that is not Unified CM, the routing script that was invoked by the ACD's dialed number must contain a TranslationRouteToVRU node to get the call to Unified CVP's Switch leg, followed by a SendToVRU node to get the call to the VoiceXML gateway and Unified CVP's VRU leg.
  - With Unified ICM 7.1 and Unified CM, the routing script that was invoked by Unified CM should use a SendToVRU node to send the call to Unified CVP using the Correlation ID. The Type10 VRU will perform an automatic second transfer to the VoiceXML gateway VRU leg.
  - All the VRU scripts that are executed by that routing script must be associated with the Type 10 Network VRU (Type 7 if Unified ICM 7.0 is used).
• For blind transfers of existing calls:
  – It does not matter to which Network VRU the Unified CVP peripheral is associated.
  – The dialed number that the ACD dialed must be associated with a Customer Instance that is associated with a Type 10 Network VRU (Type 7 if Unified ICM 7.0 is used).
  – The routing script that was invoked by the ACD's dialed number must contain a SendToVRU node to get the call to the VoiceXML gateway and Unified CVP's VRU leg.
  – All the VRU scripts that are executed by that routing script must be associated with the Type 10 Network VRU (Type 7 if Unified ICM 7.0 is used).

When Unified ICM chooses an agent or ACD destination label for a call, it tries to find one that lists a routing client that can accept that label. For calls originated by an ACD or Unified CM which are not blind transfers of existing calls, the only possible routing client is the ACD or Unified CM. Once the call has been transferred to Unified CVP, because of the handoff operation, the only possible routing client is the Unified CVP Switch leg. But in the case of blind transfers of existing calls, two routing clients are possible: (1) the Unified CVP Call Server switch leg that delivered the original call, or (2) the ACD or Unified CM. For calls that originate through Unified CVP, you can prioritize Unified CVP labels above ACD or Unified CM labels by checking the Network Transfer Preferred box in the Unified ICM Setup screen for the Unified CVP peripheral.

When using Unified CVP to do network transfers, an agent blind-transfers the caller to a new destination and the Network Transfer Preferred option is used. In this scenario, the agent should use the CTI Agent Desktop (and not the phone itself) to invoke the transfers. In addition to the CTI Agent Desktop, the Unified ICM Dialed Number Plan should be used. If configured with the same DN as the CTI Route Point, the Unified ICM Dialed Number Plan causes Unified ICM to intercept the transfer and run the Unified ICM routing script without sending the transfer commands to Unified CM through JTAPI. When the Unified ICM script returns a label, that label will be returned to the Network routing client (Unified CVP), and the caller is sent directly to the new destination. This configuration avoids a timing problem that can occur if an agent uses Unified CM CTI Route Points to initiate a network transfer.

Using Third-Party VRUs

A third-party TDM VRU can be used in any of the following ways:
• As the initial routing client (using the GED-125 Call Routing Interface)
• As a traditional VRU (using the GED-125 Call Routing Interface)
• As a Service Control VRU (using the GED-125 Service Control Interface)

In the first and second cases, the VRU acts exactly like an ACD, as described in the section on Deployment Models and Sizing Implications for Calls Originated by Cisco Unified Communications Manager and ACDs, page 5-13. Like an ACD, the VRU can be a destination for calls that arrive from another source. Calls can even be translation-routed to such devices in order to carry call context information. (This operation is known as a traditional translation route, not a TranslationRouteToVRU.) Also like an ACD, the VRU can issue its own Route Requests and invoke routing scripts to transfer the call to subsequent destinations or even to Unified CVP for self-service operations. Such transfers almost always use the Translation Route transfer mechanism.

In the third case, the VRU takes the place of either Unified CVP's Switch leg or Unified CVP's VRU leg, or it can even take the place of Unified CVP entirely. Such deployments are beyond the scope of this document.
Calls Originated by Cisco Unified Communications Manager

This chapter covers the following topics:
- Differences in Calls Originated by Cisco Unified Communications Manager, page 6-1
- Customer Call Flows, page 6-2
- Protocol Call Flows, page 6-3
- Deployment Implications, page 6-6

Differences in Calls Originated by Cisco Unified Communications Manager

A call originated by Cisco Unified Communications Manager (Unified CM) first enters the Unified ICM system when someone dials a Unified CM route point that is associated with the JTAPI interface into Unified ICM. Such calls initiate a Unified ICM routing script that can be used to place the caller into queue or into a self-service application, select an available agent, invoke Application Gateway transactions, and so forth. A call invoked through the JTAPI interface to Unified ICM is a typical post-route request; it provides a dialed number, ANI, variables, and so forth, and returns a label. Unified CM then delivers the call to the destination specified by the returned label. As with other ACD post-route requests, the exchange ends there. Unified ICM has no ability to send a subsequent label to that Unified CM unless Unified CM issues another post-route request.

This limitation is the first difference between calls originated by Unified CM and calls originated through a Unified CVP ingress gateway. Unified CVP can continue to route and re-route the call as many times as necessary. For this reason, when calls are originated from Unified CM, routing client responsibilities should be handed off to Unified CVP as soon as possible.

The second difference has to do with how calls are transferred to a VRU. ACD routing clients such as Unified CM may be transferred only by using a TranslationRouteToVRU label. When Unified CVP is the routing client, it can handle Translation Route labels as well as the Correlation ID labels that are generated by SendToVRU nodes.

The next sections provide more details on these differences.
Customer Call Flows

The following types of calls are originated by Unified CM and must be treated differently than calls originated by Unified CVP:

- Unified ICM Outbound Calls with Transfer to IVR, page 6-2
- Internal Help Desk Calls, page 6-2
- Warm Consultative Transfers and Conferences, page 6-2

Unified ICM Outbound Calls with Transfer to IVR

The Cisco Unified CCE Outbound Dialer introduces an outbound call by impersonating a Skinny Client Control Protocol (SCCP) phone and asking Unified CM to place the outbound call. When it detects that a person has answered, it transfers the call to a Unified CCE destination, taking itself out of the loop. If the customer requirement is to provide a Unified CVP message or a self-service application to the called party, then the call is transferred to Unified CVP using a Unified CM route point. This process fits the definition of a call originated by Unified CM.

Internal Help Desk Calls

Enterprises that place IP phones on employees' desks often want to provide those employees with the capability to call into a self-service application. An example might be an application that allows employees to sign up for health benefits. Or the employee might be trying to reach an agent, such as the IT help desk, and ends up waiting in queue. Both of these scenarios result in calls originating from Unified CM to Unified CVP.

The internal caller could also dial into a self-service application hosted on a VoiceXML Server that is deployed using Model #1, Standalone Self-Service. No ICM is involved in this scenario, but it still qualifies as a call originated by Unified CM.

Warm Consultative Transfers and Conferences

In a typical contact center call flow, most companies want to provide their agents with the ability to transfer callers to a second agent, who might or might not currently be available. There are two ways to do this transfer: blind transfer or warm consultative transfer (or conference).

In a blind transfer, the first agent dials a number and hangs up; the caller then gets connected to the second agent or placed into a queue if necessary. This type of transfer does not involve a call originated by Unified CM, and it is called Network Transfer. Network Transfer is also discussed in the section on ICM Managed Transfers, page 10-4.

In a warm transfer or conference, the agent dials a number and is connected to the second agent while the caller is placed on hold. The two agents can talk, then they can conference in the caller, and the first agent can then drop off. If the second agent is not available, it is the first agent (and not the caller) who is placed into a queue. All of this processing can take place without involving Unified CVP, unless the first agent needs to be queued. In that case, the first agent's call must be transferred to Unified CVP, thus creating a call originated by Unified CM.

Note

Multicast Music on Hold is not supported by Unified CVP.
Protocol Call Flows

This section describes the protocol-level call flows for calls originated by Unified CM in each of the following relevant deployment models:

- Model #1: Standalone Self-Service, page 6-3
- Model #2: Call Director, page 6-3
- Model #3a: Comprehensive Using ICM Micro-Apps, page 6-5
- Model #3b: Comprehensive Using CVP VoiceXML Server, page 6-6

Model #4, VRU Only with NIC Controlled Routing, is not discussed here because there is no NIC involved with calls originated by Unified CM.

Model #1: Standalone Self-Service

Model #1 does not involve Unified ICM. It arises when a Unified CM user dials a directory number that connects to a Unified CVP VoiceXML gateway and invokes a Unified CVP VoiceXML Server application. The VoiceXML gateway is configured in Unified CM as an H.323 gateway or SIP trunk. The call flow for this model is as follows:

1. A caller dials a route pattern.
2. Unified CM directs the call to the VoiceXML gateway.
3. The VoiceXML gateway invokes a voice browser session based on the configured Unified CVP self-service application.
4. The Unified CVP self-service application makes an HTTP request to the VoiceXML Server.
5. The VoiceXML Server starts a self-service application.
7. The caller hangs up.

The script must not execute a Transfer node, unless it is to a TDM destination. Transfers to an IP destination will result in an IP-to-IP call, which is not supported.

Model #2: Call Director

Model #2 has no VRU leg; it is all switching. Therefore, calls originated by Unified CM will always be delivered directly to their targets or else rejected. No queuing or self-service is involved.

This model assumes that the call is truly originating from Unified CM. This model excludes calls that originally arrived through a Unified CVP ingress gateway and were transferred to Unified CM, and are now being transferred again. Such situations are rare because Unified CM can usually handle those transfers itself. There are exceptions, however, such as when the target is an ACD other than Unified CM, but those situations are not covered here.
This model requires that the following items be configured:

- Unified CM route point that invokes a Unified ICM script
- Unified CVP configured as a Type 2 Network VRU
- VRU translation routes to Unified CVP
- Translation route Dialed Number Identification Service (DNIS) numbers configured in the Unified CVP Call Server
- Unified CM configured with an H.323 trunk or SIP trunk
- An H.323 gatekeeper trunk with MTP enabled, configured in Unified CM
- Unified CM route patterns for Translation Route DNIS

The call flow for this model is as follows:

1. A caller dials a route point.
2. Unified ICM invokes a routing script.
3. The routing script encounters a TranslationRouteToVRU node to transfer the call to Unified CVP. (Unified CVP is configured as a Type 2 Network VRU.)
4. Unified ICM returns the translation route label to Unified CM.
5. Unified CM consults the gatekeeper, DNS SRV, or SIP Proxy to locate the Unified CVP Call Server.
6. Unified CM connects the call to the Unified CVP Call Server.
7. Unified CM briefly establishes a G.711 media connection between the caller and the Unified CVP H.323 Service (for H.323 only).
8. The routing script encounters a Select or Label node, and it selects a target label.
9. Unified ICM returns the target label to the Unified CVP Call Server (not to the device that issued the route request).
10. The Unified CVP Call Server consults the gatekeeper, DNS SRV, or SIP Proxy to locate the destination device.
11. The Unified CVP Call Server communicates via H.323 or SIP with the target device and instructs Unified CM to establish a media stream to it.

Now consider what happens if the target device issues another route request to Unified ICM. This part of the call flow would not be possible without the initial TranslationRouteToVRU mentioned step 3.

13. The routing script encounters a Select or Label node, and it selects a target label.
14. Unified ICM returns the target label to the Unified CVP Call Server (not to the device that issued the route request).
15. The Unified CVP Call Server consults the gatekeeper, DNS SRV, or SIP Proxy to locate the destination device.
16. The Unified CVP Call Server communicates via H.323 or SIP with the target device and instructs Unified CM to establish a media stream to the device.
Model #3a: Comprehensive Using ICM Micro-Apps

Model #3a involves both call switching and VRU activity. It differs from Model #2, therefore, in that calls must be transferred to the Unified CVP VoiceXML gateway after they are transferred to the Unified CVP Switch leg. Queuing is possible in this model, as is basic prompt-and-collect activity.

This model requires that the following items be configured:

- Unified CM CTI route point that invokes a Unified ICM script
- Unified CVP configured as a Type 10 NetworkVRU
- The CTI route point configured in Unified ICM as a DN with a Type 10 NetworkVRU
- The NetworkVRU must have labels for the Unified CVP Switch leg routing client
- The NetworkVRU labels must be configured in a gatekeeper or SIP Proxy to point to VoiceXML gateways
- Unified CM configured with an H.323 trunk or SIP trunk

The call flow for this model is as follows:

1. A caller dials a route point.
2. Unified ICM invokes a routing script.
3. The routing script encounters a SendToVRU node to transfer the call to Unified CVP. (Unified CVP is configured as a Type 10 NetworkVRU.)
4. Unified ICM returns the VRU label with Correlation ID to Unified CM.
5. Unified CM consults the gatekeeper, DNS SRV, or SIP Proxy to locate the Unified CVP Call Server.
6. The call is connected to the Unified CVP Call Server.
7. Unified CM briefly establishes a G.711 media connection between the caller and the Unified CVP H.323 Service (for H.323 only).
8. Unified ICM sends a VRU transfer label with Correlation ID to the Unified CVP Call Server.
9. The Unified CVP Call Server consults the gatekeeper, DNS SRV, or SIP Proxy to locate the VoiceXML gateway.
10. The Unified CVP Call Server communicates via H.323 or SIP with the VoiceXML gateway and instructs Unified CM to establish a media stream to it.
11. The routing script executes one or more Unified CVP Microapplications via RunExternalScript nodes, plays media files, requests DTMF input, and so forth.
12. While the Unified CVP Microapplications are in progress, a target agent becomes available to take the call.
13. Unified ICM determines a label for the target agent.
14. Unified ICM returns the target label to the Unified CVP Call Server.
15. The Unified CVP Call Server consults the gatekeeper, DNS SRV, or SIP Proxy to locate the destination device.
16. The Unified CVP Call Server communicates via H.323 or SIP with the target device and instructs Unified CM to establish a media stream to it, removing the VoiceXML gateway's media stream.
If the target device later issues another route request to Unified ICM, the call flow is again exactly as described above. The call must again be transferred with Correlation ID via SendToVRU to the Unified CVP Call Server and VoiceXML gateway to create the VRU leg. Microapplications might be executed, and eventually the new target label is delivered to the Unified CVP Switch leg, which transfers the call to that target.

**Model #3b: Comprehensive Using CVP VoiceXML Server**

Model #3b does not differ significantly from Model #3a as far as call control and signaling are concerned. The only difference is that the Unified CVP Microapplications executed in Model #3b might include subdialog requests to the Unified CVP VoiceXML Server as well. Of course, self-service applications are not likely to be invoked during the period when the call is queued. Any agent selection nodes or queue nodes in the Unified ICM routing script would therefore likely be postponed until after the self-service application has completed and control has returned to the Unified ICM routing script.

**Deployment Implications**

This section presents guidelines for the following aspects of incorporating calls originated by Unified CM into the deployment:

- Unified ICM Configuration, page 6-6
- Hosted Implementations, page 6-7
- Cisco Unified Communications Manager Configuration, page 6-7
- Sizing, page 6-7

**Unified ICM Configuration**

- With Cisco Unified ICM 7.0, if you want Unified CVP to be able to perform subsequent call control, always translation-route the call to Unified CVP as a Type 2 NetworkVRU before delivering the call to its next destination. This practice creates a hand-off, putting Unified CVP in charge of subsequent transfers for the call because Unified CM is not able to receive any further labels.
- If you want to perform any queuing treatment, prompt and collect, or self-service applications, always follow the above translation route with a SendToVRU node. SendToVRU can sometimes be invoked implicitly by a Queue node or a RunExternalScript node, but you should not rely on that method. Always include an actual SendToVRU node.
- With Cisco Unified ICM 7.1, if you want Unified CVP to be able to perform subsequent call control, a translation route is not necessary if you use a Type 10 NetworkVRU. The Type 10 VRU uses the Correlation ID mechanism to perform a transfer from Unified CM to Unified CVP using a SendToVRU node. When the SendToVRU node is used with a Type 10 VRU, an initial transfer to Unified CVP hands off call control to Unified CVP, and then an automatic second transfer to the VRU leg is performed to deliver the call to a VoiceXML gateway for IVR treatment.

**Note**

This call flow and all others in this document assume Cisco Unified ICM 7.0(0) or later.

- For additional configuration requirements, see Protocol Call Flows, page 6-3.
Hosted Implementations

Translation routes sent by one ICM router must be received by a peripheral that is connected to the same ICM router. Therefore, you can translation-route a call from a Unified CM at the CICM level into Unified CVP only if Unified CVP is also located at the CICM level. In Hosted environments, this means you must provision Unified CVP Call Servers at the CICM level even if you already have other Unified CVP Call Servers at the NAM level. VoiceXML gateways and gatekeepers can, of course, be shared.

For more details on this subject, see the chapter on Interactions with Cisco Unified ICM, page 5-1.

Cisco Unified Communications Manager Configuration

The following guidelines apply to Unified CM configuration:

- Configure a gatekeeper-controlled H.323 trunk or SIP trunk. With Unified CVP 4.1 and later releases, an MTP is no longer required for H.323 trunks. Configure the gatekeeper to send calls to Unified CM using this trunk.
- Configure the appropriate route patterns for the Translation Route DNIS or VRU Label with Correlation ID appended. The Correlation ID method is used with a Type 10 VRU, and the route pattern in Unified CM must be configured to allow the extra digits to be appended, such as adding a ! to the end of the route pattern.
- When configuring agent labels, consider which device is the routing client. For cases where the label will be returned directly to Unified CM, Unified CM must be the routing client. For cases where the label will be sent to Unified CVP, the labels must be associated with each of the Unified CVP Switch leg Call Servers.

Gatekeeper or SIP Proxy Dial-Plan Configuration

If you are using a gatekeeper or SIP Proxy, the VRU label associated with the Unified CM routing client must be different than the VRU label associated with the Unified CVP routing clients. This is because the VRU label for a call originated by Unified CM is intended to send the call to the Unified CVP Call Server to hand off call control first, whereas the VRU label for a call where Unified CVP is already the routing client is intended to be sent to the VXML gateway for treatment. Once the call has been sent to Unified CVP to hand off call control, Unified CVP does a subsequent transfer to the VRU label associated with the Unified CVP routing client and delivers the call to the VXML gateway for treatment.

The dial plan in your gatekeeper or SIP Proxy should be structured as follows:

- [Unified CM routing client VRU label + correlation-id]: pointing to CVP server(s)
- [CVP routing client VRU label + correlation-id]: pointing to VXML gateway(s)

Sizing

Most customer implementations are not primarily designed for calls originated by Unified CM. The main driver is usually incoming customer calls, although calls originated by Unified CM are frequently a component, particularly in the case of warm transfers. Remember to consider those calls when sizing equipment.
Gateways

Calls originated by Unified CM do not use ingress gateways at all. Calls are delivered directly from Unified CM to the Unified CVP Call Server. They do, however, use VoiceXML gateways whenever a VRU leg is in use. Therefore, for the purposes of sizing VoiceXML gateways, consider each Unified CM call that is either in queue or conducting self-service activities.

MTP Resources

With Unified CVP 4.1 and later releases, an MTP is no longer needed for calls originated by Unified CM H.323 trunks.

The Unified CM SIP trunk allocates an MTP resource when it determines the endpoints have not negotiated a common DTMF method. If one endpoint supports RFC 2833 and the other supports out-of-band KPML events, the SIP trunk allocates an MTP resource to convert the DTMF tones. For video calls, this might cause loss of video media if the MTP resource allocated does not support video passthrough. The IPVMS-based software MTP resource does not currently support video passthrough; however, a Cisco IOS Software MTP resource can be configured to support these calls.

Unified CVP requires the SIP trunk to be set for RFC 2833 due to one-way audio issues found in testing, as well as a lack of support for KPML on the Radvision IVP component. The Video Conference Bridge does not support RFC 2833 DTMF, and some of the IP phones (such as the Cisco Unified IP Phone 7985) do not support RFC 2833 DTMF either. Therefore, it is possible to experience video loss with such endpoints in use if the correct MTP resource is not configured for use on Unified CM.
PSTN Gateway

In this type of deployment, the voice gateway is used as the PSTN voice gateway. The voice gateway is responsible for converting TDM speech to IP and for recognizing DTMF digits and converting them to H.245 or RFC2833 events. Unified CVP does not currently support passing SIP-Notify or KPML DTMF events.

In a centralized Unified CVP deployment, you can separate the VoiceXML functionality from the ingress gateway to provide a separate PSTN ingress layer. The separate PSTN layer and VoiceXML farm enables the deployment to support a large number of VoiceXML sessions and PSTN interfaces. For example, the Cisco AS5400XM can accept a DS3 connection, providing support for up to 648 DSOs. However, a gateway that is handling that many ingress calls cannot also support as many VoiceXML sessions. In such cases, the VoiceXML sessions should be off-loaded to a separate farm of VoiceXML-only gateways.

---

Note

Any TDM interface can be used as long as it is supported by the Cisco IOS gateway and by the Cisco IOS version compatible with CVP.
VoiceXML Gateway with DTMF or ASR/TTS

A standalone VoiceXML gateway is a voice gateway with no PSTN interfaces. The VoiceXML gateway enables customers to interact with the Cisco IOS VoiceXML Browser via DTMF tones or ASR/TTS. Because the gateway does not have PSTN interfaces, voice traffic is sent via Real-Time Transport Protocol (RTP) to the gateway, and DTMF tones are sent via out-of-band H.245 or RFC2833 events.

A voice gateway deployment using VoiceXML with DTMF or ASR/TTS, but no PSTN, enables you to increase the scale of the deployment and support hundreds of VoiceXML sessions per voice gateway.

In a centralized Unified CVP deployment, you could use a VoiceXML farm. For example, if you want to support 300 to 10,000 or more VoiceXML sessions, the recommended voice gateway is the Cisco AS5350XM. The standalone AS5350XM can support many DTMF or ASR/TTS VoiceXML sessions per voice gateway. In addition, Cisco recommends that you stack the AS5350XM gateways to support large VoiceXML IVR farms.

In a distributed Unified CVP deployment, consider providing an extra layer of redundancy at the branch office. You can deploy a separate PSTN gateway and a VoiceXML gateway to provide an extra layer of redundancy. In addition, for a centralized Cisco Unified Communications Manager deployment, you must provide support for Survivable Remote Site Telephony (SRST). The Cisco 3800, 3700, and 2800 Series routers are the best choice for the voice gateway because they support SRST.

VoiceXML and PSTN Gateway with DTMF or ASR/TTS

The most popular voice gateway is the combination VoiceXML and PSTN Interface Gateway. In addition, for a centralized Cisco Unified Communications Manager deployment, you must provide support for Survivable Remote Site Telephony (SRST). The Cisco 3800, 3700, and 2800 Series routers are the best choice for the voice gateway because they support SRST.

TDM Interfaces

The Cisco AS5400XM Universal Gateway offers unparalleled capacity in only two rack units (2 RUs) and provides best-of-class voice, fax, and remote-access services. High density (up to one Channelized T3 (CT3) of voice over IP (VoIP) and two CT3s of time-division multiplexing (TDM) switching), low power consumption (as low as 2.4 A at 48 VDC per G.711 CT3), high-density packet voice digital signal processor (DSP) modules, universal port DSPs, and session border control (SBC) features make the Cisco AS5400XM Universal Gateway ideal for many network deployment architectures, especially co-location environments and mega points of presence (POPs).

The Cisco AS5350XM Universal Gateway is the one-rack-unit (1 RU) gateway that supports 2-, 4-, 8-, or 16-port T1/12-port E1 configurations and provides universal port data, voice, and fax services on any port at any time. The Cisco AS5350XM Universal Gateway offers high performance and high reliability in a compact, modular design. This cost-effective platform is ideally suited for internet service providers (ISPs) and enterprise companies that require innovative universal services.
The Cisco 2600, 2800, 3700, and 3800 Series Routers support the widest range of packet telephony-based voice interfaces and signaling protocols within the industry, providing connectivity support for more than 90 percent of the world’s private branch exchanges (PBXs) and public switched telephone network (PSTN) connection points. Signaling support includes T1/E1 Primary Rate Interface (PRI), T1 channel associated signaling (CAS), E1-R2, T1/E1 QSIG Protocol, T1 Feature Group D (FGD), Basic Rate Interface (BRI), foreign exchange office (FXO), E&M, and foreign exchange station (FXS). The Cisco 2600, 2800, 3700 and 3800 Series Routers can be configured to support from two to 540 voice channels.

For the most current information about the various digital (T1/E1) and analog interfaces supported by the various voice gateways, refer to the latest product documentation available at the following sites:

- Cisco 2800 Series
- Cisco 3700 Series
- Cisco 3800 Series
- Cisco AS5300
- Cisco AS5400HPX

**Cisco Unified Border Element**

The Cisco Unified Border Element (formerly known as the Cisco Multiservice IP-to-IP Gateway) is a session border controller (SBC) that provides connectivity between IP voice networks using H.323 or SIP. A Unified Border Element is typically needed when replacing a TDM voice circuit with an IP voice trunk, such as a SIP trunk, from a telephone company. The Cisco Unified Border Element serves as a feature-rich demarcation point for connecting enterprises to service providers over IP voice trunks.

The Cisco Unified Border Element has been tested with, and can be used in, any of the following scenarios:

- SIP-to-SIP connectivity between a third-party SIP device and Cisco Unified CVP.
- SIP-to-H.323 connectivity between a third-party SIP device and Cisco Unified CVP.
- SIP-to-SIP connectivity between Cisco Unified Communications Manager and Cisco Unified CVP.
- H.323-to-H.323 connectivity between Cisco Unified Communications Manager and Cisco Unified CVP.
- Co-residency of the VoiceXML Gateway and the Cisco Unified Border Element for any of the above scenarios.
- Transcoding between G.711, G.729, and iLBC.
Gateway Choices

Unified CVP uses gateways for two purposes: TDM ingress and VoiceXML rendering. Any Cisco gateway that is supported by Unified CVP can usually be used for either purpose or both. However, depending on your deployment model, you might need only one of the functions:

- Model #1: Standalone Self-Service
  All calls use both ingress and VoiceXML.
- Model #2: Call Director
  All calls use ingress only.
- Model #3a: Comprehensive Using ICM Micro-Apps
  All calls use ingress, and some calls use VoiceXML.
- Model #3b: Comprehensive Using CVP VoiceXML Server
  All calls use ingress, and some calls use VoiceXML.
- Model #4: VRU Only with NIC Controlled Routing
  All calls use both ingress and VoiceXML.

In cases where both ingress and VoiceXML are required, you can choose to run both functions on the same gateways or you can choose to designate some gateways for ingress and others for VoiceXML. Use the following guidelines to determine whether the functions should be combined or split:

- In classical branch office deployments, where the call needs to be queued at the branch where it arrived, ingress and VoiceXML functions must always be combined.
- In cases where a large number of non-CVP PSTN connections will share the gateways, it is recommended to dedicated Ingress for that purpose, and use separate VXML gateways.
- The Cisco AS5850eRSC can be used for ingress only; it cannot be used for VoiceXML.
- VoiceXML-only gateways are less costly because they do not require DSP farms or TDM cards. Use a spreadsheet to determine which way you obtain the best price.
• With relatively low call volume, it is usually better to combine the functions for redundancy purposes. Two combined gateways are better than one of each because the loss of one gateway still allows calls to be processed, though at a lower capacity.

The next decision is whether to use Cisco Integrated Service Router (ISR) gateways (Cisco 2800 or 3800 Series) or the Cisco AS5x00 Series Gateways. Guidelines state that ISR gateways should be used only in branch office sites, whereas AS5x00 Series gateways should be used in centralized data center sites. Sometimes it is difficult to determine what constitutes a branch office, and therefore which gateway should be used. The following guidelines can help with that determination:

• The classical definition of branch offices, for which you should use ISR gateways, includes:
  – Multiple sites where TDM calls will be arriving from the PSTN.
  – Those sites are separated from the data centers where most of the Unified CVP equipment resides.
  – You will be placing only one gateway at each site.
• If you have sites where you will be stacking multiple gateways for any reason, then those sites are data center sites and should use Cisco AS5x00 Series gateways.

For more information on the Cisco AS5x00 Series Gateways, refer to the technical specifications available at


For more information on the Cisco Integrated Service Routers (ISRs), refer to the documentation available at


**Gateway Sizing**

Individual Cisco gateways can handle various call capacities depending on whether they are doing ingress only, VoiceXML only, or a combination of the two. Gateways doing VoiceXML activities also have different call capacities depending on whether or not they are supporting ASR or TTS activities.

In general, gateways performing ingress-only can be sized according to the number of TDM cables that can be connected to them. For gateways that are combined or VoiceXML-only, it is important to ensure that the overall CPU usage will be less than 70% on average. The numbers in Table 7-1 are based on Unified CVP VoiceXML documents; other applications that generate more complex VoiceXML documents will have a higher impact on performance. The following factors affect CPU usage:

• Calls per second (cps)
• Maximum concurrent calls
• Maximum concurrent VoiceXML sessions

Before sizing the voice gateways, use the IPCC Resource Calculator to determine the maximum number of trunks (DS0s) and VoiceXML IVR ports needed to support the entire solution.

For almost all Unified CVP deployment models, sizing is based on the maximum number of concurrent VoiceXML sessions and VoIP calls, as listed in Table 7-1 and Table 7-2.
Chapter 7      Gateway Options

Gateway Sizing

Table 7-1 Maximum Number of VoiceXML Sessions Supported by Cisco Voice Gateways

<table>
<thead>
<tr>
<th>Cisco Voice Gateway Platform</th>
<th>Dedicated VoiceXML Server</th>
<th>Voice Gateway and VoiceXML</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>VoiceXML and DTMF</td>
<td>VoiceXML and ASR/TTS</td>
</tr>
<tr>
<td>2801</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>2811</td>
<td>30</td>
<td>24</td>
</tr>
<tr>
<td>2821</td>
<td>48</td>
<td>36</td>
</tr>
<tr>
<td>2851</td>
<td>60</td>
<td>56</td>
</tr>
<tr>
<td>3725</td>
<td>68</td>
<td>50</td>
</tr>
<tr>
<td>3745</td>
<td>100</td>
<td>80</td>
</tr>
<tr>
<td>3825</td>
<td>120</td>
<td>96</td>
</tr>
<tr>
<td>3845</td>
<td>150</td>
<td>144</td>
</tr>
<tr>
<td>AS5400HPX</td>
<td>96</td>
<td>90</td>
</tr>
<tr>
<td>AS5350XM</td>
<td>240</td>
<td>192</td>
</tr>
<tr>
<td>AS5400XM</td>
<td>240</td>
<td>192</td>
</tr>
</tbody>
</table>

The numbers in Table 7-1 assume that the only activities running on these gateways are VoiceXML with basic routing and connectivity. If you intend to run additional applications, such as fax, security, normal business calls, and so forth, then the capacity numbers presented here should be prorated accordingly. These figures apply to Cisco IOS Release 12.4 (Mainline).

**Note**

These performance numbers are accurate when used with either the Cisco Call Server or Cisco VoiceXML Server. Performance can, and often does, vary with different applications. Performance from external VoiceXML applications (such as Nuance OSDMs) might not be representative of the performance when interoperating with non-Cisco applications. You must ensure that the CPU usage is less than 70% on average and that adequate memory is available on Cisco gateways at full load when running external VoiceXML applications. Users should contact the application provider of the desired VoiceXML application for performance and availability information. Be aware that external VoiceXML applications are not provided by Cisco, and Cisco makes no claims or warranties regarding the performance, stability, or feature capabilities of the application when interoperating in a Cisco environment.

**Note**

If you run VoiceXML on one of the Cisco 2800, 3700, or 3800 gateways, additional licenses (FL-VXML-1 or FL-VXML-12) are required.

Also consult Table 7-2 to ensure that the concurrent call load and call arrival rates do not exceed the listed capacities.
In addition to these capacities, also consider how much DRAM and flash memory to order. The capacity that comes with the machine by default is usually sufficient for most purposes. However, if your application requires large numbers of distinct .wav files (as with complex self-service applications) or if your application has unusually large .wav files (as with extended voice messages or music files), you might want to increase the amount of DRAM in order to accommodate more cache space. The .wav files are recorded at 8 kbps. Additionally, if you plan to use the flash memory itself rather than a media server to house media files, you might want to expand your flash memory order. The use of DRAM for prompt caching is discussed in detail in the chapter on Media File Options, page 12-1.

### Table 7-2 Maximum Number of Calls Supported per Gateway Platform

<table>
<thead>
<tr>
<th>Cisco Voice Gateway Platform</th>
<th>Maximum Calls per Second</th>
<th>Maximum VoIP Calls</th>
</tr>
</thead>
<tbody>
<tr>
<td>2801</td>
<td>1</td>
<td>32</td>
</tr>
<tr>
<td>2811</td>
<td>1</td>
<td>80</td>
</tr>
<tr>
<td>2821</td>
<td>1.2</td>
<td>128</td>
</tr>
<tr>
<td>2851</td>
<td>2</td>
<td>192</td>
</tr>
<tr>
<td>3725</td>
<td>2</td>
<td>192</td>
</tr>
<tr>
<td>3745</td>
<td>4</td>
<td>384</td>
</tr>
<tr>
<td>3825</td>
<td>4</td>
<td>384</td>
</tr>
<tr>
<td>3845</td>
<td>8</td>
<td>540</td>
</tr>
<tr>
<td>AS5400HPX</td>
<td>7</td>
<td>384</td>
</tr>
<tr>
<td>AS5350XM</td>
<td>20</td>
<td>192</td>
</tr>
<tr>
<td>AS5400XM</td>
<td>20</td>
<td>648</td>
</tr>
</tbody>
</table>

Using MGCP Gateways

Cisco Unified CVP requires the deployment of H.323 or SIP gateways. However, customers might require the use of MGCP 0.1 voice gateways with Cisco Unified Communications Manager deployments for purposes of overlap sending, NSF, and Q.SIG support. The following design considerations apply to deploying Cisco Unified CVP in this environment:

- Design and plan a phased migration of each MGCP voice gateway to H.323 or SIP.
- Implement both MGCP 0.1 and H.323 or SIP.
  - Because of the way MGCP works, a PSTN interface using MGCP can be used for MGCP only. Therefore, if you want to use MGCP for regular Cisco Unified Communications Manager calls and H.323 or SIP for Unified CVP calls, you will need two PSTN circuits.
- Deploy a second H.323 or SIP voice gateway at each Unified CVP location.
- Send calls through the Cisco Unified Communications Manager to Unified CVP.
When sending calls through Cisco Unified Communications Manager to Unified CVP, the following guidelines apply:

- The Unified CVP survivability.tcl script cannot be used in this solution. If the remote site is disconnected from the central site, the call will be dropped.

- There will be an additional hit on the performance of Cisco Unified Communications Manager. This is because, in a "normal" Unified CVP deployment, Cisco Unified Communications Manager resources are not used until the call is sent to the agent. In this model, Cisco Unified Communications Manager resources are used for all calls to Unified CVP, even if they terminate in self-service. This is in addition to the calls that are extended to agents. If all calls are eventually extended to an agent, the performance impact on Cisco Unified Communications Manager is approximately double that of a "normal" Unified CVP deployment. This factor alone typically limits this scenario to small call centers only.

- In order to queue calls at the edge, you must use the sigdigits feature in Unified CVP to ensure that the calls are queued at the appropriate site or VXML gateway. For more information on how the sigdigits feature works, see the chapters on Distributed Deployments, page 3-1, and Designing Unified CVP for High Availability, page 4-1.
CHAPTER 8

Design Implications for VoiceXML Server

This chapter covers the following topics:

- What is VoiceXML over HTTP?, page 8-1
- Multi-Language Support, page 8-2
- Differences in the Supported Web Application Servers, page 8-2
- Where to Install Unified Call Studio, page 8-3

What is VoiceXML over HTTP?

Communication between the VoiceXML (VXML) server and the Voice Browser is based on request-response cycles using VXML over HTTP. VXML documents are linked together by using the Uniform Resource Identifiers (URI), a standardized technology to reference resources within a network. User input is carried out by web forms similar to HTML. Therefore, forms contain input fields that are edited by the user and sent back to a server.

Resources for the Voice Browser are located on the VXML server. These resources are VXML files, digital audio, instructions for speech recognition (Grammars) and scripts. Every Communication process between the VXML browser and Voice Application has to be initiated by the VXML browser as a request to the VXML server. For this purpose, VXML files contain Grammars which specify expected words and phrases. A Link contains the URL that refers to the Voice application. The browser connects to that URL as soon as it recovers a match between spoken input and one of the Grammars.

When gauging VXML server performance, consider the following key aspects:

- QoS and network bandwidth between the Web application server and the voice gateway
  
  See the section on Bandwidth Provisioning and QoS Considerations, page 9-1, for more details.

- Performance on the VXML Server
  

- Use of prerecorded audio versus Text-to-Speech (TTS)
  
  Voice user-interface applications tend to use prerecorded audio files wherever possible. Recorded audio sounds much better than TTS. Prerecorded audio file quality must be designed so that it does not impact download time and browser interpretation. Make recordings in 8-bit mu-law 8 kHz format.
Audio file caching
Make sure the voice gateway is set to cache audio content to prevent delays from having to download files from the media source. For more details about prompt management on supported gateways, see Configuring Caching and Streaming in Cisco IOS, page 12-3.

Use of grammars
A voice application, like any user-centric application, is prone to certain problems that might be discovered only through formal usability testing or observation of the application in use. Poor speech recognition accuracy is one type of problem common to voice applications, and a problem most often caused by poor grammar implementation. When users mispronounce words or say things that the grammar designer does not expect, the recognizer cannot match their input against the grammar. Poorly designed grammars containing many difficult-to-distinguish entries also results in many mis-recognized inputs, leading to decreased performance on the VXML server. Grammar tuning is the process of improving recognition accuracy by modifying a grammar based on an analysis of its performance.

Multi-Language Support
The Cisco IOS Voice Browser or the Media Resource Control Protocol (MRCP) specification does not impose restrictions on support for multiple languages. However, there might be restrictions on the automatic speech recognition (ASR) or TTS server. Check with your preferred ASR/TTS vendor about their support for your languages before preparing a multilingual application.

You can dynamically change the ASR server value by using the command cisco property com.cisco.asr-server in the VXML script. This property overrides any previous value set by the VXML script.

Differences in the Supported Web Application Servers
From a very high-level perspective, IBM WebSphere Application Server (http://www.ibm.com/websphere) is a complete J2EE application server environment complete with an administration console and connection pooling. However, Tomcat (http://tomcat.apache.org/) is a simple and basic environment with a Servlet Engine and a Java Server Pages engine only. The decision to use Tomcat or WebSphere Application Server depends on your current enterprise infrastructure requirements. In many cases, Tomcat is more than sufficient. But if you already have WebSphere infrastructure and management capabilities or have a preference for WebSphere in general, you should use it for Unified CVP.

Performance tests conducted on the web application server showed only slight variations in the processor performance between the two Web Application Servers using metrics such as the following:

- Impact of call volume
- Impact of application size
- Impact of application complexity

Either a Tomcat or WebSphere Application server running Unified CVP VXML can support up to 750 simultaneous calls per Cisco MCS-7845 server.
Where to Install Unified Call Studio

Cisco Unified Call Studio is an Integrated Development Environment (IDE). As in the case of any IDE, the Unified Call Studio needs to be installed in a setup that is conducive for development, such as workstations that are used for other software development or business analysis purposes. Because the Unified Call Studio is Eclipse-based, many other development activities (such as writing Java programs or building object models) can be migrated to this tool so that developers and analysts have one common utility for most of their development needs.

Because the Unified Call Studio has not been tested with Microsoft Windows 2003, Cisco does not recommended co-locating the Call Studio with the VXML Server.
Network Infrastructure Considerations

This chapter presents deployment characteristics and provisioning requirements of the Unified CVP network. Provisioning guidelines are presented for network traffic flows between remote components over the WAN, including recommendations for applying proper Quality of Service (QoS) to WAN traffic flows.

For the most current information on network considerations, refer to the sections on deployment models, bandwidth, and QoS presented in the latest version of the Cisco Unified Contact Center Enterprise Solution Reference Network Design (SRND), available at

http://www.cisco.com/go/srnd

This chapter covers the following topics:

- Bandwidth Provisioning and QoS Considerations, page 9-1
- Bandwidth Sizing, page 9-4
- Call Admission Control, page 9-8
- QoS Marking, page 9-9
- Blocking Initial G.711 Media Burst, page 9-9
- Network Security Using Firewalls, page 9-10

Bandwidth Provisioning and QoS Considerations

In many Unified CVP deployments, all components are centralized; therefore, there is no WAN network traffic to consider. In general, there are only two scenarios when WAN network structure must be considered in a Unified CVP environment:

- In a distributed Unified CVP deployment, when the ingress gateways are separated from the Unified CVP servers by a WAN.
- In Unified CVP deployments where the ingress gateway and the agent are separated over a WAN. The agent can be a TDM ACD agent or a Unified CCE agent.

Unlike Unified ICM, Unified CVP has a very simple view of QoS:

- Unified CVP has no concept of a private WAN network structure. All WAN activity, when required, is conducted on a converged WAN network structure.
- Unified CVP does not use separate IP addresses for high and low priority traffic.
- Unified CVP does mark the QoS DSCP of SIP packets. H.323 traffic must be marked by routers or switches in the network using access control lists (ACLs).
Adequate bandwidth provisioning is an important component in the success of Unified CVP deployments. Bandwidth guidelines and examples are provided in this chapter to help with provisioning the required bandwidth.

**Unified CVP Network Architecture Overview**

In a Unified CVP environment, WAN and LAN traffic can be grouped into the following categories:

- **Voice Traffic**, page 9-2
- **Call Control Traffic**, page 9-2
- **Data Traffic**, page 9-4

**Voice Traffic**

Voice calls consist of Real-Time Transport Protocol (RTP) packets that contain actual voice samples. RTP packets are transmitted in the following cases:

- **Between the ingress PSTN gateway or originating IP phone and one of the following:**
  - Another IP phone, such as an agent
    The destination phone might or might not be co-located with the ingress gateway or calling IP phone, and the connection can be over a WAN or LAN.
  - An egress gateway front-ending a TDM ACD (for legacy ACDs or IVRs)
    The egress gateway might or might not be co-located with the ingress gateway, and the connection can be over a WAN or LAN.
  - A VoiceXML gateway performing prompt-and-collect treatment
    The VoiceXML gateway will usually be the same gateway as the ingress gateway, but it can be different. In either case, both the ingress and VoiceXML gateways are typically co-located (located on the same LAN). The connection is typically over a LAN but can be over a WAN.
- **Between the VoiceXML gateway and the ASR/TTS server.** The RTP stream between the VoiceXML gateway and ASR/TTS server must be G.711, and the connection can be over a WAN or LAN.

**Call Control Traffic**

There are several types of call control traffic in a Unified CVP solution. Call control functions include those used to set up, maintain, tear down, or redirect calls.

**H.323 or SIP**

Unified CVP is currently certified with three types of VoIP endpoints: Cisco IOS voice gateways, Cisco Unified Communications Manager (Unified CM), and the PGW in either Call Control mode or Signaling mode. Call Control traffic flows between the following endpoints:

- **Ingress gateway to/from Unified CVP Call Server**
  The ingress gateway can be a PGW, Unified CM, or a Cisco IOS gateway, or other SIP device in the case of SIP. The connection can be over a WAN or LAN.
Unified CVP Call Server to/from egress gateway

The egress gateway can be Unified CM or a Cisco IOS gateway. The egress gateway is either a VoiceXML gateway used to provide prompt-and-collect treatment to the caller, or it is the target of a transfer to an agent (Unified CCE or TDM) or a legacy TDM IVR. The connection can be over a WAN or LAN.

GED-125

The Unified CVP Call Server and the Unified ICM VRU PG communicate using the GED-125 protocol. The GED-125 protocol includes:

- Messages that control the caller experience, such as notification when a call arrives
- Instructions to transfer or disconnect the caller
- Instructions that control the IVR treatment the caller experiences

The VRU PG normally connects to Unified CVP over a LAN connection. However, in deployments that use clustering over the WAN, it is possible for Unified CVP to connect to the redundant VRU PG across the WAN.

At this time, no tool exists that specifically addresses communications between the VRU PG and Unified CVP. However, bandwidth consumed between the Unified ICM Central Controller and VRU PG is very similar to the bandwidth consumed between the VRU PG and Unified CVP.

The VRU Peripheral Gateway to ICM Central Controller Bandwidth Calculator tool is available (with proper login authentication) through the Cisco Steps to Success Portal at


You can also access the Bandwidth Calculator directly (with proper login authentication) at


If the VRU PGs are split across the WAN, the total bandwidth required would be double what the calculator tool reports: once for Unified ICM Central Controller to VRU PG and once for VRU PG to Unified CVP.

Media Resource Control Protocol (MRCP)

The VoiceXML gateway communicates with ASR/TTS servers using Media Resource Control Protocol (MRCP) v1.0. This protocol currently works with Real-Time Streaming Protocol (RTSP) to help establish control connections to ASR/TTS servers such as Nuance, Scansoft, and IBM WebSphere Voice Server. The connection can be over the LAN or WAN.

ICM Central Controller to Unified CVP VRU PG

No tool exists that specifically addresses communications between the Unified ICM Central Controller and the Unified CVP VRU PG. Testing has shown, however, that the tool for calculating bandwidth needed between the Unified ICM Central Controller and the IP IVR PG also produces accurate measurements for Unified CVP if you perform the following substitution in one field:

For the field labeled Average number of RUN VRU SCRIPT nodes, substitute the number of Unified ICM script nodes that interact with Unified CVP. Nodes that can interact with Unified CVP are Run External Script, Label, Divert Label, Queue to Skill Group, Queue to Agent, Agent, Release, Send to VRU, and Translation Route to VRU.

This bandwidth calculator tool is available (with proper login authentication) at:


The connection in this case can be over a WAN or LAN.
Data Traffic

Data traffic includes VoiceXML documents and prerecorded media files returned as a result of HTTP requests executed by the VoiceXML gateway. Specifically:

- The VoiceXML gateway requests media files in an HTTP request to a media file server. The media server response returns the media file in the body of the HTTP message. The VoiceXML gateway then converts the media files to RTP packets and plays them to the caller. The connection in this case can be over a WAN or LAN.
- The VoiceXML gateway requests VoiceXML documents from either the Unified CVP VoiceXML Server or the Unified CVP IVR Service. The connection in this case can be over a WAN or LAN.

This chapter focuses primarily on the types of data flows and bandwidth used between a remote ingress gateway and the components with which it interfaces:

- Unified CVP VoiceXML Server
- Unified CVP Call Server IVR Service
- Unified CVP Call Server SIP or H.323 Service
- IP phones
- Media servers
- Egress gateways
- ASR or TTS servers

Guidelines and examples are presented to help estimate required bandwidth and, where applicable, provision QoS for these network segments.

Bandwidth Sizing

As discussed above, most of the bandwidth requirements in a Unified CVP solution occur in a Distributed Unified CVP topology, due primarily to the fact that the ingress and/or VoiceXML gateway is separated from the servers that provide it with media files, VoiceXML documents, and call control signaling. For purposes of the following discussion, assume all calls to a branch begin with one minute of IVR treatment followed by a single transfer to an agent that also lasts one minute. Each branch has 20 agents, and each agent handles 30 calls per hour for a total of 600 calls per hour per branch. The call average rate is therefore 0.166 calls per second (cps) per branch.

Note that even a slight change in these variables might have a large impact on sizing. It is important to remember that .166 calls per second is an average for the entire hour. Typically, calls do not come in uniformly across an entire hour, and there are usually peaks and valleys within the busy hour. Try to find the busiest traffic period, and calculate the call arrival rate based on the worst-case scenario.

VoiceXML Documents

VoiceXML (VXML) documents are generated based on voice application scripts written using either Unified ICM scripts or Unified CVP VoiceXML Studio, or both. A VoiceXML document is generated for every prompt that is played to the caller. The VoiceXML documents vary in size, depending on the type of prompt being used; menu prompts with many selections are much larger than a simple prompt that simply plays an announcement.
On average, a VoiceXML document between the Unified CVP Call Server or VXML Server and the gateway is about 7 kilobytes. You can calculate the bandwidth used by approximating the number of prompts that will be used per call, per minute. The calculation is as follows:

\[
7,000 \text{ bytes} \times 8 \text{ bits} = 56,000 \text{ bits per prompt}
\]

\[
56,000 \times \frac{\text{(Number of prompts per minute)}}{60} = \text{kbps per port}
\]

However, if you are going to use a more complex application that uses many menu prompts (more than the average estimated above) or if you want to calculate the bandwidth more exactly, you can use the VoiceXML document sizes listed in Table 9-1 to calculate the amount of bandwidth needed. The document sizes in Table 9-1 are measured from the VXML Server to the VXML Gateway.

**Table 9-1 Approximate Size of VXML Document Types**

<table>
<thead>
<tr>
<th>VXML Document Type</th>
<th>VXML Document Size (approximate)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root document (one required at beginning of call)</td>
<td>19,000 bytes</td>
</tr>
<tr>
<td>Subdialog_start (at least one per call at beginning of call)</td>
<td>700 bytes</td>
</tr>
<tr>
<td>Query gateway for Call-ID and GUID (one required per call)</td>
<td>1,300 bytes</td>
</tr>
<tr>
<td>Menu (increases in size with number of menu choices)</td>
<td>1,000 bytes + 2,000 bytes per menu choice</td>
</tr>
<tr>
<td>Play announcement (simple .wav file)</td>
<td>1,100 bytes</td>
</tr>
<tr>
<td>Cleanup (one required at end of call)</td>
<td>4,000 bytes</td>
</tr>
</tbody>
</table>

**Media File Retrieval**

Media files (prompts) can be stored locally in flash memory on each router. This method eliminates bandwidth considerations, but maintainability becomes an issue because a prompt that requires changes must then be replaced on every router. If the prompts are instead stored on an HTTP media server (or an HTTP cache engine), the gateway can locally cache voice prompts once it has initially retrieved the prompts. If configured correctly, the HTTP media server can cache many, if not all, prompts, depending of the number and size of the prompts. The refresh period for the prompts is defined on the HTTP media server. Therefore, the bandwidth utilized would be limited to the initial load of the prompts at each gateway, plus periodic updates after the expiration of the refresh interval.

Not caching prompts at the gateway causes significant Cisco IOS performance degradation (as much as 35% to 40%) in addition to the extra bandwidth usage. For the most current information on configuring gateway prompt caching, refer to the latest version of the *Configuration and Administration Guide for Cisco Unified Customer Voice Portal (CVP)*, available at


Assume that there is a total of 50 prompts, with an average size of 50 kB and a refresh interval of 15 minutes. The bandwidth usage would then be:

\[
(50 \text{ prompts}) \times (50,000 \text{ bytes/prompt}) \times (8 \text{ bits/byte}) = 20,000,000 \text{ bits}
\]

\[
(20,000,000 \text{ bits}) / (900 \text{ secs}) = 22.2 \text{ average kbps per branch}
\]
**H.323 Signaling**

Every call that is processed by the branch gateway requires 6000 bytes, plus 1000 bytes for each transferred call to an agent, giving a total of 56,000 bits per call (7000 bytes * 8 bits). Thus, the average bandwidth required would be \((0.166 \times 56 \text{ kbps}) = 9.3 \text{ kbps}\) for the WAN link to the remote branch.

**SIP Signaling**

SIP is a text-based protocol, therefore the packets used are larger than with H.323. The typical SIP call flow uses about 17,000 bytes per call. Using the previous bandwidth formulas based on calls per second, the average bandwidth usage would be:

\[
(17,000 \text{ bytes/call}) \times (8 \text{ bits/byte}) = 136,000 \text{ bits per call}
\]

\[
(0.166 \text{ calls/second}) \times (136 \text{ kilobits/call}) = 22.5 \text{ average kbps per branch}
\]

**ASR and TTS**

Centralized Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) are now supported in distributed Unified CVP deployments as of Unified CVP 4.0. In order to support this model, QoS must be configured on the network and bandwidth must be reserved specifically for the ASR/TTS RTP and MRCP traffic. ASR/TTS cannot use silence suppression and must use the G.711 codec, therefore centralized ASR/TTS is bandwidth intensive. ASR/TTS RTP and MRCP traffic is not tagged with QoS DSCP markings, therefore it is necessary to use access control lists (ACLs) to classify and re-mark the traffic at the remote site and central site.

**Classifying RTP Media Traffic Between VoiceXML Gateways and ASR/TTS Servers**

The RTP port range used by the VoiceXML gateway is the normal Cisco IOS RTP UDP port range of 16384 to 32767; however, the RTP UDP port range used by the ASR/TTS server can vary by OS and ASR/TTS vendor. It is possible to construct an ACL to match the traffic from the ASR/TTS server based on the VoiceXML gateway UDP port range; but if possible, Cisco recommends finding the ports used by the ASR/TTS server as well. The RTP traffic should be marked with DSCP EF so that it is placed in the priority queue with other voice traffic.

The QoS priority queue must also be configured to support the maximum number of ASR/TTS sessions anticipated. If a call admission control mechanism such as Cisco Unified CM locations or Resource Reservation Protocol (RSVP) is used, this extra priority queue bandwidth should not be included when configuring the locations or RSVP bandwidth. For example, if you want to support two ASR/TTS G.711 sessions (80 kbps each) as well as four IP telephony phone calls using G.729 (24 kbps each), the priority queue total bandwidth would be 256 kbps. The locations call admission control or RSVP bandwidth should be limited to only the IP telephony bandwidth (96 kbps in this example). Configuring the locations or RSVP bandwidth with 256 kbps would allow IP telephony calls to use all of the bandwidth and conflict with the ASR/TTS sessions.
Classifying MRCP Traffic Between VoiceXML Gateways and ASR/TTS Servers

The MRCP traffic is much easier to classify. ASR/TTS servers listen on TCP 554 for MRCP requests, therefore this port should be used in ACLs to classify the traffic. The bandwidth used by MRCP can vary depending on how often the application uses the ASR/TTS resource. MRCP uses about 2000 bytes per interaction. If there is an ASR/TTS interaction every 3 seconds per call, you can calculate the average bandwidth as follows:

\[
\text{(2000 bytes/interaction)} \times \text{ (20 interactions/minute)} \times \text{ (8 bits/byte)} = 320,000 \text{ bits per minute per call}
\]

\[
\frac{320,000 \text{ bits per minute}}{60 \text{ seconds/minute}} = 5.3 \text{ average kbps per branch}
\]

If you configure a maximum of 6 ASR/TTS sessions at any given time, then (6 \times 5.3 \text{ kbps}) = 32 average kbps per branch.

Limiting the Maximum Number of ASR/TTS-Enabled Calls

It is possible to limit the number of calls enable for ASR/TTS so that, once the limit is reached, regular DTMF prompt-and-collect can be used instead of rejecting the call altogether. In the following example, assume 5559000 is the ASR/TTS DNIS and 5559001 is the DTMF DNIS. You can configure the ingress gateway to do the ASR load limiting for you by changing the DNIS when you have exceeded maximum connections allowed on the ASR/TTS VoIP dial peer.

```
voice translation-rule 3
  rule 3 /5559000/ /5559001/
  !
voice translation-profile change
  translate called 3
  !
  !Primary dial-peer is ASR/TTS enabled DNIS in ICM script
dial-peer voice 9000 voip
  max-conn 6
  preference 1
  destination-pattern 55590..
  ...
  !
  !As soon as 'max-conn' is exceeded, next preferred dial-peer will change the DNIS to a
  DTMF prompt & collect ICM script
dial-peer voice 9001 voip
  translation-profile outgoing change
  preference 2
  destination-pattern 55590..
  ...
  !
```

**Note**

80 kbps is the rate for G.711 full-duplex with no VAD, including IP/RTP headers and no compression.
24 kbps is the rate for G.729 full-duplex with no VAD, including IP/RTP headers and no compression.
Voice Traffic

Unified CVP can support both G.711 and G.729. G.711 must be used for the VoiceXML portion of the call, but G.729 can be used once the call is connected to an Agent. If you are using ASR/TTS for speech recognition, then G.711 must be used because ASR/TTS servers support only G.711. For the most current bandwidth information on voice RTP streams, refer to the latest version of the Cisco Unified Communications SRND Based on Cisco Unified Communications Manager, available at http://cisco.com/go/srnd

Call Admission Control

Call admission control is the mechanism for determining if there is enough bandwidth available on the network to carry an RTP stream. Unified CM can use its own locations mechanism or RSVP to track bandwidth between the ingress gateway and destination IP phone locations.

For more information about call admission control, see the chapter on Distributed Deployments, page 3-1.

RSVP

Cisco Unified CM 5.0 introduced support for Resource Reservation Protocol (RSVP) between endpoints within a cluster. RSVP is a protocol used for call admission control, and it is used by the routers in the network to reserve bandwidth for calls. RSVP can be used for delivering calls to Unified CCE agents in a Unified CM cluster. For more information on RSVP, refer to the latest version of the Cisco Unified Communications SRND Based on Cisco Unified Communications Manager, available at http://cisco.com/go/srnd
QoS Marking

The Unified CVP Call Server marks only the QoS DSCP for SIP messages. If QoS is needed for Unified
CVP H.323 signaling and data traffic across a WAN, configure network routers for QoS using the IP
address and ports to classify and mark the traffic as recommended in Table 9-2.

Table 9-2   Recommended Port Usage and QoS Settings

<table>
<thead>
<tr>
<th>Component</th>
<th>Port</th>
<th>Queue</th>
<th>PHB</th>
<th>DSCP</th>
<th>Maximum Latency (Round Trip)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Media Server</td>
<td>TCP 80</td>
<td>CVP-Data</td>
<td>AF11</td>
<td>10^1</td>
<td>1 sec</td>
</tr>
<tr>
<td>Unified CVP Call Server, H.323</td>
<td>TCP 1720</td>
<td>Call Signaling</td>
<td>CS3</td>
<td>24</td>
<td>200 ms</td>
</tr>
<tr>
<td>Unified CVP Call Server, SIP</td>
<td>TCP or UDP 5060</td>
<td>Call Signaling</td>
<td>CS3</td>
<td>24</td>
<td>200 ms</td>
</tr>
<tr>
<td>Unified CVP IVR Service</td>
<td>TCP 8000</td>
<td>CVP-Data</td>
<td>AF11</td>
<td>10^1</td>
<td>1 sec</td>
</tr>
<tr>
<td>Unified CVP VoiceXML Server</td>
<td>TCP 7000</td>
<td>CVP-Data</td>
<td>AF11</td>
<td>10^1</td>
<td>1 sec</td>
</tr>
<tr>
<td>Ingress Gateway, H.323</td>
<td>TCP 1720</td>
<td>Call Signaling</td>
<td>CS3</td>
<td>24</td>
<td>200 ms</td>
</tr>
<tr>
<td>Ingress Gateway, SIP</td>
<td>TCP or UDP 5060</td>
<td>Call Signaling</td>
<td>CS3</td>
<td>24</td>
<td>200 ms</td>
</tr>
<tr>
<td>VoiceXML Gateway, H.323</td>
<td>TCP 1720</td>
<td>Call Signaling</td>
<td>CS3</td>
<td>24</td>
<td>200 ms</td>
</tr>
<tr>
<td>VoiceXML Gateway, SIP</td>
<td>TCP or UDP 5060</td>
<td>Call Signaling</td>
<td>CS3</td>
<td>24</td>
<td>200 ms</td>
</tr>
<tr>
<td>H.323 Gatekeeper</td>
<td>UDP 1719</td>
<td>Call Signaling</td>
<td>CS3</td>
<td>24</td>
<td>200 ms</td>
</tr>
<tr>
<td>SIP Proxy Server</td>
<td>TCP or UDP 5060</td>
<td>Call Signaling</td>
<td>CS3</td>
<td>24</td>
<td>200 ms</td>
</tr>
<tr>
<td>MRCP</td>
<td>TCP 554</td>
<td>Call Signaling</td>
<td>CS3</td>
<td>24</td>
<td>200 ms</td>
</tr>
</tbody>
</table>

1. The DSCP (or PHB) value for CVP-Data traffic is only a recommendation. You can choose the actual DSCP value used to mark the traffic according to your preference.

Neither the CVP-Data queue nor the Signaling queue is a priority queue as described in Cisco IOS router terminology. The priority queue is used for voice or other real-time traffic, while call signaling and Unified CVP traffic are reserved a certain amount of bandwidth based on the call volume.

Blocking Initial G.711 Media Burst

When a gateway first receives a call, the gateway signals the Unified CVP Call Server using H.323 in
order to hand off the call control responsibilities. To establish this initial call, a short media stream is
established between the gateway and the Unified CVP Call Server. The media stream is only in one
direction, from the gateway to the Unified CVP Call Server. Because this media stream is not accounted
for by Unified CM’s locations-based call admission control, Cisco recommends that the media stream
be blocked from traversing bandwidth-constrained links to avoid oversubscribing the priority queue.
This precaution is needed only for H.323 deployments; SIP deployments do not have this consideration.
The following example illustrates the ACL configuration:

```
access-list 100 deny udp host 10.0.0.1 host 10.10.0.100 range 16384 32767
access-list 100 permit ip any any

interface serial0/0
ip access-group 100 out
```
In the preceding example, 10.0.0.1 is the voice gateway’s H.323-bound IP address and 10.10.0.100 is the Unified CVP Call Server. If there are multiple Call Servers, add one ACL entry for each. The interface serial0/0 is the WAN interface connecting to the central site that is hosting the Unified CVP Call Server.

**Network Security Using Firewalls**

When configuring network security using firewalls or ACLs, refer to Table 9-3 for information about TCP/UDP ports used by Unified CVP, voice gateways, VoiceXML gateways. For a complete listing of ports used by Unified CVP, refer to the *Unified CVP 7.0 Port Utilization Guide*.

<table>
<thead>
<tr>
<th>Source and Destination Component</th>
<th>Destination Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voice Gateway to Media Server</td>
<td>TCP 80</td>
</tr>
<tr>
<td>Voice Gateway to Unified CVP Call Server H.225</td>
<td>TCP 1720</td>
</tr>
<tr>
<td>Voice Gateway to Unified CVP Call Server SIP</td>
<td>TCP or UDP 5060</td>
</tr>
<tr>
<td>Voice Gateway to Unified CVP Call Server</td>
<td>TCP 8000</td>
</tr>
<tr>
<td>Voice Gateway to Unified CVP VoiceXML Server</td>
<td>TCP 7000</td>
</tr>
<tr>
<td>Voice Gateway to MRCP Server</td>
<td>TCP 554</td>
</tr>
<tr>
<td>Unified CVP Call Server to Egress Voice Gateway H.225</td>
<td>TCP 1720</td>
</tr>
<tr>
<td>Unified CVP Call Server to Egress Voice Gateway SIP</td>
<td>TCP or UDP 5060</td>
</tr>
<tr>
<td>Unified CVP Call Server to VoiceXML Gateway H.225</td>
<td>TCP 1720</td>
</tr>
<tr>
<td>Unified CVP Call Server to VoiceXML Gateway SIP</td>
<td>TCP or UDP 5060</td>
</tr>
<tr>
<td>Unified CVP Call Server to H.323 Gatekeeper</td>
<td>UDP 1719</td>
</tr>
<tr>
<td>Unified CVP Call Server to SIP Proxy Server</td>
<td>TCP or UDP 5060</td>
</tr>
</tbody>
</table>
Call Transfer Options

Designing for call transfers is one of the major steps required when designing a Unified CVP deployment. There are numerous transfer options that can be used with Unified CVP. The goal of this chapter is to explain each of the various options and to provide pros, cons, and considerations associated with each.

This chapter covers the following topics:

- Release Trunk Transfers, page 10-1
- ICM Managed Transfers, page 10-4
- SIP Refer Transfer, page 10-5
- Intelligent Network (IN) Release Trunk Transfers, page 10-5
- VoiceXML Transfers, page 10-6

Release Trunk Transfers

This section deals with the types of transfers that release the ingress trunk, thus removing the gateway and Unified CVP from the call control loop. There is no tromboning in these cases. These transfers have the following characteristics:

- Release Trunk Transfers can be invoked by the VoiceXML server (standalone model) or via the Unified ICM.
- Unified ICM Network Transfer using Unified CVP as the routing client will not work because Unified CVP can no longer control the call.
- These transfers are blind, meaning that if the transfer fails for any reason, Unified ICM does not recover control of the call. Router Requery is not supported.
- From the standpoint of Unified ICM reporting, Release Trunk Transfers cause the switch leg to terminate, resulting in a TCD record being written to the database for the call even though the caller is still potentially talking to an agent. This behavior differs from other types of transfers in which the TCD record does not get finalized until the caller actually hangs up.
- Because the ingress trunk is released, you do not have to size gateways to include calls that have been transferred in this way. This behavior differs from other types of transfers in which gateway resources continue to be occupied until the caller hangs up.
- Because Unified CVP is no longer monitoring the call, you do not have to size Unified CVP Call Directors to include calls that have been transferred in this way. Additionally, Unified CVP Call Director port licenses are not required.
There are three signaling mechanisms available to trigger a release trunk transfer:

- **Takeback-and-Transfer (TNT), page 10-2**
- **Hookflash and Wink, page 10-2**
- **Two B Channel Transfer (TBCT), page 10-3**

### Takeback-and-Transfer (TNT)

TNT (also known as Transfer Connect) is a transfer mechanism offered by some U.S. PSTN service providers (such as AT&T and Verizon). With this transfer method, inband DTMF tones are outpulsed to the PSTN by Unified CVP. These inband tones act as a signaling mechanism to the PSTN to request a transfer to be completed. A typical DTMF sequence is *8xxxx, where xxxx represents a new routing label that the PSTN understands. Upon detection of a TNT DTMF sequence, the PSTN drops the call leg to the ingress gateway port and then re-routes the caller to a new PSTN location (such as a TDM ACD location).

This behavior might be necessary for a customer with existing ACD site(s) but no IVR, who wants to use Unified CVP initially as just an IVR. Over time, the customer might want to transition agents from the TDM ACD(s) to Cisco Unified CCE and use Unified CVP as an IVR, queueing point, and transfer pivot point (thus eliminating the need for TNT services).

In Unified CVP deployments with the ICM, the DTMF routing label outpulsed could have been a Unified ICM translation routing label to enable passing of call data to another Unified ICM peripheral (such as a TDM ACD). In this scenario, Unified CVP views the call as completed, and Unified CVP call control is ended. With TNT, if the transfer to the termination point fails, there is nothing Unified CVP can do to re-route the call. While some TNT services do have the ability to re-route the call back to Unified CVP, Unified CVP sees this call as a new call.

### Hookflash and Wink

Hookflash and wink are signaling mechanisms typically associated with a TDM PBX or ACD. Hookflash applies only to analog trunks and wink applies only to digital trunks (T1 or E1 channel), but otherwise they are similar in function. Both hookflash and wink send an on-hook or off-hook signal to the PBX or ACD, which responds with dial tone (or the PBX winks back on a digital trunk). This signaling causes the voice gateway to send a string of routing digits to the PBX or ACD. Upon collection of the routing digits, the PBX or ACD transfers the caller to the new termination, which could be an ACD queue or service on that same PBX or ACD.

This behavior might be necessary for a customer with an existing ACD but no IVR, who wants to use Unified CVP initially as an IVR logically installed on the line side of their existing PBX or ACD. Over time, the customer might want to transition agents from the TDM ACD to Cisco Unified CCE and have the voice gateways connected to the PSTN instead of the line side of the PBX or ACD. In Unified CVP deployments with Unified ICM, the routing label could be a Unified ICM translation routing label. This label enables passing of call data to the ACD service (and subsequently to the agent in a screen pop). With hookflash and wink, if the transfer to the termination point fails, there is nothing Unified CVP can do to re-route the call. While some PBX or ACD models do have the ability to re-route the call back to Unified CVP, Unified CVP sees this call as a new call.

Hookflash transfer has been problematic in the past because the PBXs and the gateways have constrained support for this feature. If at all possible, avoid using the PBX for Unified ICM switching, and terminate all incoming calls on Unified CVP ingress gateways rather than on the PBX, thus allowing Unified CVP to route calls to the PBX rather than the other way around.
However, if hookflash transfers are required, the following guidelines and notes apply:

- Cisco 1700 Series Gateways were not tested with hookflash transfers.
- Cisco 2800 and 3800 Series Gateways can support Analog FXO or Digital FXO (T1/CAS). This function is considered line-side hookflash to the PBX, and it worked very well in tests with Avaya Definity G3. (However, E&M is not supported at this time.) You can adjust the hookflash duration with the command `timing hookflash-out` under the `voice-port`. This feature is useful if you have a PBX that has a non-configurable hookflash duration, and it gives you the ability to adjust the hookflash duration on the gateway side.
- Cisco 5x00 Series Gateways were tested with T1/CAS and the command `e&m-fgb dtmf dnis`. E&M is considered "trunk-side hookflash" to the PBX, and not all switches support trunk-side hookflash (the Avaya Definity G3 does not). Additionally, the hookflash duration on the Cisco 5x00 Series Gateways is 200 ms, and you must configure the PBX for this same duration. This option varies with switch type, and a proof-of-concept with the switch vendor is highly recommended.
- In Deployment Model #1, Standalone Self-Service, a TCL script is required to produce the hookflash. A TCL script is provided with Unified CVP.
- In all cases, Automatic Number Identification (ANI) is not available to the call when it gets to Unified CVP. In some Unified CVP deployment models, the ICM might already know the ANI if the call had been pre-routed there.
- In all cases, Dialed Number Identification Service (DNIS) must be configured on the gateway, based on the T1/E1 channel on which the call arrives. The PBX is programmed to route certain DNIS calls over certain T1 trunks. Because the call arrives to the gateway on that trunk, you can definitively configure its DNIS. The drawback to this approach is that the gateway trunk allocation must be predetermined. You must know what percentage of calls arrive to which DNISs so that the trunk groups on the gateway can be allocated accordingly.

An alternate method that can be used on some PBXs is a "converse on step," whereby DTMF tones indicating DNIS and ANI are sent to the IVR. This method requires a single main Unified ICM routing script to input DNIS digits using a Get Data (GD) Microapplication and to invoke the correct sub-script based on the collected DNIS digits. This method requires close coordination between Cisco, the PBX vendor, and the customer, and it has not yet been tested.

### Two B Channel Transfer (TBCT)

TBCT is an ISDN-based release trunk signaling mechanism that is offered by some PSTN service providers. When a TBCT is invoked, the ingress gateway places the initial inbound call on hold briefly while a second call leg (ISDN B Channel) is used to call the termination point. When the termination point answers the call, the gateway sends ISDN signaling to the PSTN switch to request that the transfer be completed and that the call be bridged through the PSTN switch and removed from the ingress gateway. As with a TNT transfer, the termination point might be a TDM PBX or ACD connected to the PSTN.

This behavior might be necessary for a customer with existing ACD site(s) but no IVR, who wants to use Unified CVP initially as just an IVR. Over time, the customer might want to transition agents from the TDM ACD(s) to Cisco Unified CCE and use Unified CVP as an IVR, queueing point, and transfer pivot point (thus eliminating the need for TBCT services). If the call to the termination point fails, there is nothing Unified CVP can do to re-route the call.
ICM Managed Transfers

Most Unified CVP customers use Unified ICM Managed transfers. Unified CVP performs this function most naturally, providing gateway-based switching for Unified ICM and Unified CCE installations.

In Unified CVP deployments with Unified ICM, Unified ICM provides all call control. VoiceXML call control from the Unified CVP VoiceXML Server is not supported when Unified ICM is deployed with Unified CVP.

Unified ICM Managed transfers transfer the call to a new termination point, which can be any of the following:

- A Cisco Unified Communications Manager phone
- An egress port on the same gateway as the ingress port
- A distant egress gateway that has a TDM connection to a TDM ACD or PBX (making use of toll bypass features)
- A Unified CVP VoiceXML gateway for queuing or self-service activities

To terminate the call, the voice gateway selects an outgoing POTS or VoIP dial-peer based on the destination specified by Unified ICM. When a Unified ICM VoIP transfer occurs, the ingress voice gateway port is not released. If the termination point is an egress voice gateway, then a second voice gateway port is utilized. Unified CVP continues to monitor the call, and Unified ICM also retains control of the call and can instruct Unified CVP to transfer the call to a new destination.

This type of transfer is used when Unified CVP is used as a call treatment platform and queue point for Unified CCE agents. Unified CVP could also be used to provide call treatment to front-end calls to TDM ACD locations supported by Unified ICM. This type of transfer allows for calls to be transferred between peripherals supported by Unified ICM, with full call context and without any tromboning of the voice path.

Calls that are transferred in this way have the following characteristics:

- Unified ICM Network Transfer using Unified CVP as the routing client functions properly because Unified CVP continues to control the call.
- These transfers are supervised, meaning that if the transfer fails for any reason, the Unified ICM routing script does recover control via the Router Requery mechanism.
- From the standpoint of Unified ICM reporting, the switch leg does not terminate until the caller actually hangs up. Thus, the TCD record that is written for the switch leg of the call encompasses the entire life of the call, from initial ingress to hang-up.
- Because the ingress trunk is not released, you must size gateways to include calls that have been transferred in this way.
- Because Unified CVP continues to monitor the call, you must size Unified CVP Call Servers to include calls that have been transferred in this way. Additionally, Unified CVP Call Director port licenses are required, except for calls that are connected to Cisco Unified Communications Manager agents.
SIP Refer Transfer

In some scenarios, it is desirable for Unified CVP to transfer a call to a SIP destination and not have Unified ICM and Unified CVP retain any ability for further call control. Unified CVP can perform a SIP Refer transfer, which allows Unified CVP to remove itself from the call, thus freeing up licensed Unified CVP ports. The Ingress Voice Gateway port remains in use until the caller or the terminating equipment releases the call. SIP Refer transfers may be used in both Comprehensive and Call Director deployments.

A SIP Refer transfer can be invoked by either of the following methods:

- Unified ICM sends Unified CVP a routing label with a format of rfXXXX (For example, rf5551000).
- An application-controlled alternative is to set an ECC variable (user.sip.referertransfer) to the value y in the Unified ICM script, and then send that variable to Unified CVP.

The SIP Refer transfer can be invoked after Unified CVP queue treatment has been provided to a caller. SIP Refer transfers can be made to Cisco Unified Communications Manager or other SIP endpoints, such as a SIP-enabled ACD.

H.323 Refer Transfer

Unified CVP 4.0(2) introduces a new transfer mechanism for H.323 calls that behaves in a similar manner to SIP Refer. This feature allows Unified CVP to remove itself from the call, thus freeing up call control ports. Using this feature, the call can be queued at the VoiceXML gateway and then sent to an agent on Cisco Unified Communications Manager or other H.323 endpoints such as an ACD.

Unified CVP cannot execute further call control operations after this kind of transfer has been executed; however, Unified CVP Survivability can still be used for failure recovery in this scenario. This feature can be used in both Comprehensive and Call Director call flow models, and it is available only for PSTN-originated calls via a Cisco IOS gateway running the Unified CVP Survivability service.

The H.323 Refer transfer can be invoked by either of the following methods:

- Unified ICM sends Unified CVP a routing label with a format of RF88#xxxx# (For example, RF88#5551000#).
- An application-controlled alternative is to set an ECC variable (user.h323.rftransfer) to the value y in the Unified ICM script, and then send that variable to Unified CVP. The CVP H.323 service will modify the received label automatically to conform to the format given above.

The Unified CVP Survivability service should be enabled to execute the H323 Refer transfers by using the following parameter:

```
param icm-rf 1
```

Intelligent Network (IN) Release Trunk Transfers

Customers using Deployment Model #4 (VRU Only with NIC Controlled Routing) rely on call switching methods that do not involve Unified CVP. In these situations, all switching instructions are exchanged directly between a Unified ICM Network Interface Controller (NIC) and the PSTN. Examples of such NIC interfaces include Signaling System 7 (SS7) and Call Routing Service Protocol (CRSP). The SS7 NIC is also used as an interface into the PGW in deployments that involve that device. Thus, PGW deployments perform this type of transfer.
VoiceXML Transfers

VoiceXML call control is supported only in standalone Unified CVP deployments (Deployment Model #1) in which call control is provided by the VoiceXML server. Deployment Model #3b, which also incorporates the VoiceXML Server, does not support VoiceXML call control. In those and all Unified ICM integrated deployments, Unified ICM must make all call control decisions.

The VoiceXML Server can invoke three types of transfers: Release Trunk transfers, VoiceXML blind transfers, and VoiceXML bridged transfers. Release Trunk transfers result in the incoming call being released from the ingress voice gateway. VoiceXML blind transfers result in the call being bridged to an egress voice gateway or a VoIP endpoint, but the VoiceXML server releases all subsequent call control. VoiceXML bridged transfers result in the call being bridged to an egress voice gateway or a VoIP endpoint, but the VoiceXML server retains call control so that it can return a caller to an IVR application or transfer the caller to another termination point.

Release Trunk transfers from the VoiceXML server are invoked using the `subdialog_return` element. The VoiceXML server can invoke a TNT transfer, TBCT transfer, and HookFlash/Wink transfers as well as SIP Refer transfers. For TDM Release Trunk transfers (TNT, TBCT and Hookflash/Wink), the VoiceXML gateway must be combined with the ingress gateway in order for the Release Trunk transfer to work.

VoiceXML blind and bridged transfers are invoked using the Transfer element in VoiceXML Studio. VoiceXML Transfers will transfer the call to any dial-peer that is configured in the gateway.

VoiceXML Blind Transfers differ from VoiceXML Bridged Transfers in the following ways:

- VoiceXML blind transfers do not support call progress supervision, whereas Bridged transfers do. This means that if a blind transfer fails, the VoiceXML Server script does not recover control and cannot attempt a different destination or take remedial action.
- VoiceXML blind transfers cause the VoiceXML Server script to end. Always connect the "done exit" branch from a Blind transfer node to a subdialog_return and a hang-up node.

Bridged transfers do not terminate the script. The VoiceXML Server waits until either the ingress or the destination call ends. The script ends only if the ingress call leg hangs up. If the destination call leg hangs up first, the script recovers control and continues with additional self-service activity. Note that the VoiceXML Server port license remains in use for the duration of a bridged transfer, even though the script is not actually performing any processing.
11 Using the GKTMP NIC

This chapter covers the following topics:

- The Cisco Gatekeeper External Interface, page 11-1
- The Unified ICM GKTMP NIC, page 11-1
- Typical Applications of GKTMP with Unified CVP, page 11-2

The Cisco Gatekeeper External Interface

The Cisco H.323 Gatekeeper provides an external interface that uses Gatekeeper Transaction Message Protocol (GKTMP) to hand off the processing of Registration Admission Status (RAS) requests to external applications. This feature allows organizations to supplement the on-board capabilities of the gatekeeper and to provide support for externally managed dial plans and intelligent call routing in an H.323 voice network.

GKTMP is based on RAS and provides a set of ASCII request and response messages that can be used to exchange information between the Cisco IOS Gatekeeper and the external application over a TCP connection.


The Unified ICM GKTMP NIC

Using its GKTMP NIC, Unified ICM can function as a GKTMP server for the gatekeeper, processing GKTMP request messages as route requests and running routing scripts in the normal way. The RAS sourceInfo Alias and destinationInfo Alias are made available to the Unified ICM script as the Calling Line ID and Dialed Number respectively, the latter typically being used for script selection by the Unified ICM Router. The Unified ICM script might perform many different functions, including database or back-end system access, and finally sends a label back to the NIC for return to the gatekeeper and ultimately back to the requesting endpoint as the modified destinationInfo Alias. The Unified ICM script can also modify the sourceInfo Alias. However, not all requesting endpoints use the translated sourceInfo Alias that is returned; Cisco IOS gateways make use of it, whereas the Unified CVP Call Server and Cisco Unified Communications Manager both ignore it.

To force the gatekeeper to reject the Admission Request (ARQ) and return an Admission Reject (ARJ) to the requesting endpoint, the Unified ICM script can return a BUSY label, optionally with an additional reason code (for example, DESTINATION_UNKNOWN).
Typical Applications of GKTMP with Unified CVP

The GKTMP NIC can be used with Unified CVP in both pre-routing and post-routing call scenarios. The former is used to process the Admission Request from the ingress gateway before the call is routed to the Unified CVP Call Server and answered, the latter for transfers being performed by the Unified CVP Call Server.

- **Intelligent rejection before the call is answered by Unified CVP**
  
  When the Unified CVP Call Server receives an H.225 SETUP message, it answers the call by returning a CONNECT message immediately. Sometimes it is necessary to make a routing decision before delivering the call to Unified CVP and before the call is answered. One example is the use of look-ahead routing, in which the Unified ICM script determines the availability and reachability of other Unified ICM peripherals that will be required for the overall call scenario once the call has been delivered to Unified CVP. With the GKTMP NIC, it is possible to reject calls intelligently for alternate routing via the TDM network rather than answering them and not having the resources to handle them subsequently.

- **Selection of Unified CVP Call Server based on H.323 call information**
  
  Occasionally the gatekeeper static configuration is not sufficient for selection of the most appropriate Unified CVP Call Server to handle an incoming call. For example, the routing decision might need to be based on the calling line ID or source signalling address.

- **Manipulation of the calling line ID**
  
  Modification of the sourceInfo Alias is sometimes useful in order to overload the calling line ID with additional information required by the destination endpoint, where translation routing is not possible.

- **Unified ICM-based dial plan**
  
  Unified ICM implements a centralized H.323 voice network dial plan, reducing the need for dial plan configuration on individual gatekeepers. This approach is appropriate only if the dial plan is large, complex, dynamic, and difficult to maintain across multiple gatekeepers.

- **Time-of-day routing**
  
  This feature allows the gatekeeper routing to be supplemented with decisions based on date and time, possibly to handle time-dependent resource availability.

- **Back-end system and database queries**
  
  Unified ICM database lookup or application gateway capabilities data from external systems can be incorporated into routing decisions.

- **Filtering calls that might be sent immediately to an available destination and bypass Unified CVP**
  
  While this approach might be seen as a way to avoid using Unified CVP Call Server resources, it limits the functionality available. For example, there can be no intelligent re-query for alternative destinations on ring-no-answer, nor will this approach allow the call to be taken back to Unified CVP for subsequent call treatment and transfers.
• Pre-routing with context passing to Unified CVP

This method is used if call context collected during the pre-routing phase of the call needs to be passed to Unified CVP rather than simply performing a standalone routing request via the GKTMP NIC. In this example, the Unified CVP Call Server must be configured as a Type 2 VRU so that the call can be translation-routed to it and still perform a subsequent transfer.

Protocol-Level Call Flow

Fundamentally, GKTMP allows a Unified ICM routing script to provide additional external business rules that are called by the gatekeeper to select an alternative destination label or target IP address for a call, given a dialed number or label. However, the protocol-level call flow differs depending on the purpose for which the GKTMP request is being used, as described in the following scenarios:

• Pre-routing of incoming calls, but call context passing is not required, page 11-3

In this mode of operation, calls arriving at an ingress gateway make a request to Unified ICM to select either a particular Unified CVP Call Server target or a non-CVP target. When a Unified CVP Call Server is selected, the call is delivered to Unified CVP as a completely new call, with no link to the Unified ICM script involved in the GKTMP-based routing step.

• Pre-routing of incoming calls, and call context passing is required, page 11-4

In this scenario, calls that arrive at an ingress gateway make a request to Unified ICM via the GKTMP NIC before being delivered to a particular Unified CVP Call Server target. Any information obtained by this initial Unified ICM routing script is preserved and made available to the Unified ICM script as processing resumes when the call is delivered to the Unified CVP Call Server.

• Routing of post-ICM calls, page 11-4

This is to modify the routing of calls that are being transferred to a destination label returned to Unified CVP by an ICM routing script. No information obtained by the previous routing script is available to the new script invoked by the GKTMP request, which functions in a purely standalone manner.

Pre-routing of incoming calls, but call context passing is not required

1. A call arrives at the ingress gateway.
2. The ingress gateway requests the gatekeeper to identify a target Unified CVP Call Server (or other IP destination).
3. The gatekeeper issues a GKTMP Request ARQ to Unified ICM via the GKTMP NIC.
4. Unified ICM starts a routing script based on dialed number, ANI, time of day, and so forth.
5. The Unified ICM routing script might return either a Response ARQ or a Response ACF to the gatekeeper. In the former case, Unified ICM returns modified information in the response, and the gatekeeper resumes ARQ processing to select the IP endpoint. This approach is adopted if Unified ICM is returning a destination label only and not selecting the required destination IP endpoint address(ies) explicitly. In the latter case, Unified ICM completes the processing of the request, returning modified information and the selected target IP endpoints. In this case the gatekeeper regards the request as completed, does no further processing of the request, and returns the ACF to the endpoint that issued the ARQ. The Unified ICM routing script ends at this point.
6. The gatekeeper returns the selected IP address to the ingress gateway.
7. If the target is not a Unified CVP device, the ingress gateway sets up a VoIP call to that target.
8. If the target is a Unified CVP Call Server, the ingress gateway sets up a new call to it.
9. The Unified CVP Call Server sends a New Call message to Unified ICM.
10. Unified ICM starts an independent routing script to handle the incoming call.
11. Normal call flow continues. Transfer to VRU leg, Transfer to agent, as well as subsequent blind
    Network VRU Transfers to secondary agents or return-to-queue are fully supported.

Pre-routing of incoming calls, and call context passing is required
1. A call arrives at the ingress gateway.
2. The ingress gateway requests the gatekeeper to identify a target Unified CVP Call Server (or other
   IP destination).
3. The gatekeeper issues a GKTMP Request ARQ to Unified ICM via the GKTMP NIC.
4. Unified ICM starts a routing script based on dialed number, ANI, time of day, and so forth.
5. The Unified ICM routing script executes a TranslationRouteToVRU to select a target Unified CVP
   Call Server.
6. Unified ICM returns the selected translation route label (and optionally the destination endpoint IP
   address) in the GKTMP Response ARQ or ACF via the GKTMP NIC.
7. The gatekeeper returns the selected IP address to the ingress gateway.
8. The ingress gateway sets up a new call to the selected Unified CVP Call Server.
9. The Unified CVP Call Server sends a RequestInstruction message to Unified ICM.
10. The Unified ICM routing script resumes after the TranslationRouteToVRU node.
11. Normal call flow continues. Transfer to VRU leg, Transfer to agent, as well as subsequent blind
    Network VRU Transfers to secondary agents or return-to-queue are fully supported. (For limitations
    in Unified ICM versions prior to 7.0(0), see Deployment Implications, page 11-5.)

Routing of post-ICM calls
1. Unified ICM selects a target agent or other destination label. The Unified ICM routing script ends
   at this point.
2. Unified ICM returns the selected label to the Unified CVP Call Server.
3. The Unified CVP Call Server requests the gatekeeper for the endpoint IP address associated with
   that label.
4. The gatekeeper issues a GKTMP Request ARQ to Unified ICM via the GKTMP NIC.
5. Unified ICM starts a completely independent routing script based on the selected label, ANI, time
   of day, and so forth.
6. This new Unified ICM routing script selects an appropriate target for the call.
7. Unified ICM returns the selected label (and optionally the destination endpoint IP address) in the
   GKTMP Response ARQ or ACF response via the GKTMP NIC. The Unified ICM routing script
   ends at this point.
8. The gatekeeper returns the selected endpoint IP address to the Unified CVP Call Server.
9. The Unified CVP Call Server performs and Empty Capability Set transfer, communicating with the
    ingress gateway and the transfer destination endpoint to establish a VoIP call between them.
Deployment Implications

GKTMP is a simple request/response protocol. From the perspective of Unified ICM, this means that the GKTMP NIC cannot perform any call control other than returning a single label and/or endpoint IP address. Third-party call control through the GKTMP NIC is not possible once that single destination has been returned. However, it is possible when call control responsibilities are handed off to Unified CVP. The following discussion covers these call control implications for each of the three types of call flows described in the previous section:

- **Pre-routing of incoming calls, but call context passing is not required, page 11-5**
- **Pre-routing of incoming calls, and call context passing is required, page 11-5**
- **Routing of post-ICM calls, page 11-5**
- **Other implications, page 11-5**

**Pre-routing of incoming calls, but call context passing is not required**

Two ICM routing scripts are required for this option. The first script identifies the initial target for the incoming call; the second script is the normal routing script for call handling at Unified CVP, which is described elsewhere in this guide. The transfer from one script to the other is via a plain label; it is not a translation route or a VRU leg transfer. The GKTMP-initiated script cannot perform any queuing or other VRU activity; it can only modify the Admission Request content and optionally return an endpoint IP address.

In the case where the pre-routing script returns a label that is associated with an endpoint other than a Unified CVP Call Server, no further call control is possible unless the endpoint is an ACD or VRU with its own post-routing interface into Unified ICM and its own ability to perform call control operations.

**Pre-routing of incoming calls, and call context passing is required**

With this option, the GKTMP-initiated and Unified CVP-initiated routing scripts are one and the same. A TranslationRouteToVRU node must be used to move the call to a Type 2 Unified CVP NetworkVRU. This node must precede any Queue node if the customer needs the ability to perform any subsequent agent-to-agent transfers, even if an appropriate agent is already available. This action might seem like an extraneous transfer, but it is necessary in order to force call control hand-off to Unified CVP.

Following the TranslationRouteToVRU and prior to execution of any RunExternalScript nodes, a second VRU transfer is required using a SendToVRU node to establish the VRU call leg to the VoiceXML gateway.

**Routing of post-ICM calls**

Two completely independent Unified ICM routing scripts are used in this scenario. The only connection between the two is that the label returned by the first routing script becomes a Dialed Number that the gatekeeper uses to invoke the second routing script. This transfer is not via a translation route or VRU leg, and call context is not available to the second script, other than what is included in the gatekeeper request itself.

**Other implications**

Note that inserting the GKTMP NIC into the call flow does result in additional route requests being processed by the Unified ICM Router for each call processed in this way, and additional call detail records are written by the Unified ICM Logger. Where possible, configure the GKTMP server triggers on the gatekeeper so that only those calls specifically requiring the additional routing functionality afforded by the GKTMP NIC generate a Request ARQ to Unified ICM.
Media File Options

This chapter covers the following topics:

- Deployment and Ongoing Management, page 12-1
- Co-Resident Call Server, Media Server, and VoiceXML Server, page 12-1
- Bandwidth Calculation for Prompt Retrieval, page 12-3
- Configuring Caching and Streaming in Cisco IOS, page 12-3
- Configuring Caching and Streaming in Cisco IOS, page 12-3
- Branch Office Implications, page 12-6

Deployment and Ongoing Management

Voice prompts can be stored in the following locations:

- In flash memory on each local gateway
  In this way, gateways do not have to retrieve .wav files for prompts, so WAN bandwidth is not affected. However, if a prompt needs to change, you must change it on every gateway. Cisco recommends that you store prompts in flash only for critical prompts such as error messages or other messages that can be used when the WAN is down.

- On an HTTP media server
  In this way, each local gateway (if properly configured) can cache many or all prompts, depending on the number and size of the prompts (up to 100 MB of prompts). The best way to test whether your media server is appropriately serving the media files is to use a regular web browser such as Internet Explorer and specify the URL of a prompt on the media server, such as http://10.4.33.130/en-us/sys/1.wav. Your web browser should be able to download and play the .wav file without any authentication required.

Co-Resident Call Server, Media Server, and VoiceXML Server

If your Call Server, Media Server, and VoiceXML (VXML) Server reside on the same hardware server and you have multiple co-resident servers, Unified CVP will not automatically use the same physical server for call control, VXML, and media file services. Just because the components are co-resident, that does not force one component to use the other co-resident components, and it is just as likely to use the components located on another server.
By default, the components are load-balanced across all of the physical servers and do not attempt to use the same server for all of the services. Over the course of thousands of calls, all of the components on all of the servers will be load-balanced and equally utilized, but for one particular call it is possible to be using several different physical servers. Because of this, for one particular call you can be using H.323/SIP call control on one server, VoiceXML on another server, and the media files on yet another server.

You can simplify management and troubleshooting by configuring Unified CVP to use the same physical server for all of these functions on a per-call basis. Of course, if there is only one server in the system, then this is not a concern. The instructions below show you how to configure Unified CVP so that it does use components on the same physical server instead of load-balancing and using a random server for each component.

Perform the following steps to choose the co-resident VoiceXML (VXML) Server in the ICM Script Editor:

1. When setting up the media_server ECC variable that specifies your VXML Server in the ICM script, use the Formula Editor to set the media_server ECC variable to http://call.routingClient:7000/CVP, where call.routingClient is the built-in call variable that ICM sets automatically for you. The routing client name in ICM is not necessarily the same as the Unified CVP Server's hostname (and usually is not the same).

2. You can then use the routing client name as a hostname in the VXML gateway:
   
   ip host cvp_rc_1 10.4.33.130

3. Configure the routing client hostname for every Unified CVP Server Routing Client.

Perform the following steps to choose the co-resident Media Server in Call Studio:

1. In the ICM script, set one of the ToExtVXML[] array variables with the call.routingclient data, such as "ServerName=call.routingclient." This variable will be passed to the Unified CVP VXML Server, and the variable will be stored in the session data with the variable name ServerName.

2. In Call Studio, use a substitution to populate the Default Audio Path. Add the Application_Modifier element found under the Context folder, and specify the Default Audio Path under the Settings tab in the following format:

   http://{Data.Session.ServerName}

   If you are using Micro-Apps in conjunction with the VXML Server, you will have to pay careful attention to the media_server ECC variable in the ICM script because the same variable is used to specify both the VXML Server and the Media Server, but the contents of the variable uses a different format depending on which server you want to specify. The media_server ECC variable should be used as indicated below whenever you want to use a Micro-App for prompting. If you subsequently want to use the VXML Server, you will have to rewrite this variable by following the instructions above.

   1. When setting up the media_server ECC variable that specifies your Media server in the ICM script, use the Formula Editor to set the media_server ECC variable to http://call.routingClient, where call.routingClient is the built-in call variable that ICM sets automatically for you. The routing client name in ICM is not necessarily the same as the Unified CVP Server's hostname (and usually is not the same).

   2. You can then use the name of the routing client as a hostname in the VXML gateway:

      ip host cvp_rc_1 10.4.33.130

   3. Configure the routing client hostname for every Unified CVP Server Routing Client.
Bandwidth Calculation for Prompt Retrieval

When prompts are stored on an HTTP media server, the refresh period for the prompts is defined on that server. The bandwidth consumed by prompts consists of the initial loading of the prompts at each gateway and the periodic updates at the expiration of the refresh interval.

As an example of determining the bandwidth consumed by prompts, assume that a deployment has 50 prompts with an average size of 50 kB (50,000 bytes) each. Also assume that the refresh period for the prompts is defined as 15 minutes (900 seconds) on the HTTP media server. The WAN bandwidth required for prompts in this deployment can be calculated as follows:

\[
\text{(50 prompts)} \times (50,000 \text{ bytes/prompt}) \times (8 \text{ bits/byte}) = 20,000,000 \text{ bits}
\]

\[
\frac{20,000,000 \text{ bits}}{900 \text{ seconds}} = 22.2 \text{ kbps per branch}
\]

Configuring Caching and Streaming in Cisco IOS

The Cisco IOS VoiceXML Browser uses an HTTP client, which is a part of Cisco IOS. The client fetches VoiceXML documents, audio files, and other file resources. There are two key properties associated with playing audio prompts: caching and streaming. These two properties are closely related to each other, and they can affect system performance greatly when the router is under load.

Streaming and Non-Streaming

In non-streaming mode, the entire audio file must be downloaded from the HTTP server onto the router before the Media Player can start playing the prompt. This implies delay for the caller. If the audio file is relatively small, the caller should not notice any delay because downloading a small file should take only a few milliseconds. The delay of loading larger files can be overcome by using either caching or streaming mode.

In streaming mode, the Media Player "streams" the audio in "media chunks" from the HTTP server to the caller. As soon as the first chunk is fetched from the server, the Media Player can start playing. The advantage of streaming mode is that there is no noticeable delay to the caller, irrespective of the size of the audio prompt. The disadvantage of streaming mode is that, because of all of the back-and-forth interactions from fetching the media file in chunks, it deteriorates performance. Additionally, the ability to cache the files in memory reduces the advantage of streaming large files directly from the HTTP server.

The recommendation for a Unified CVP VoiceXML gateway is to use non-streaming mode for the prompts in combination with caching. The Cisco IOS command to configure non-streaming mode is:

```
ivr prompt streamed none
```

Caching

There are two types of cache involved in storing media files: the IVR Media Player cache and the HTTP Client cache. The HTTP Client cache is used for storing files that are downloaded from the HTTP server. In non-streaming mode, the entire media file is stored inside the HTTP Client cache. In streaming mode, the first chunk of the media file is stored in the HTTP Client cache and in the IVR cache, and all subsequent chunks of the file are saved in the IVR cache only.
Because of the above recommendation to use only non-streaming mode, the IVR prompt cache is never used and the HTTP Client cache is the primary cache. The HTTP Client cache also has the advantage of being able to store 100 MB of prompts, whereas the IVR cache is limited to 16 MB.

To configure the HTTP Client cache, use the following IOS commands:

```
http client cache memory file <1-10000>
```

Where `<1-10000>` is the file size in kilobytes. The default maximum file size is 50 kB, but the recommended file size is 600 kB. Any file that is larger than the configured HTTP Client memory file size will not be cached.

```
http client cache memory pool <0-100000>
```

Where `<0-100000>` is the total memory size available for all prompts, expressed in kilobytes. A value of zero disables HTTP caching. The default memory pool size for the HTTP Client cache is 10 Mb. The recommended memory pool size is the total size of all prompts stored on the media server, up to 100 MB.

### Caching Query URLs

A query is a URL that has a question mark (`?`) followed by one or more "name=value" attribute pairs in it. The CVP VoiceXML server uses query URLs heavily when generating the dynamic VoiceXML pages that are rendered to the caller. Because each call is unique, data retrieved from a query URL is both wasteful of cache memory and a possible security risk because the query URL can contain information such as account numbers or PINs.

Query URL caching is disabled by default in Cisco IOS. To ensure that it is disabled, issue a `show run` command in Cisco IOS and ensure that the following Cisco IOS command does not appear:

```
http client cache query
```

### TCP Socket Persistence

The overhead for opening and closing the TCP socket connections can take a toll on the system performance, especially when the applications issue many small requests one after another. To reduce this socket connection overhead, the client can keep the socket open after a previous application request is fulfilled, so that the next application can reuse the same connection. This is feasible as long as the two connections have the same host IP address and port number. This kind of connection is referred to as a **persistent connection**. As the name implies, the connection can last over a long period of time without being shut down.

To establish a persistent connection, both the client and the server must agree that the connection is going to be a persistent one. To configure the Cisco IOS HTTP Client to request a persistent connection from the server, configure the following command:

```
http client connection persistent
```

### Cache Aging

The HTTP Client manages its cache by the "freshness" of each cached entry. Whether a cached entry is fresh or stale depends on two numbers: Age and FreshTime. Age is the elapsed time since the file was last downloaded from the server. FreshTime is the duration that the file is expected to stay fresh in the HTTP Client cache since the file was last downloaded.
There are several variables that can affect the FreshTime of a file, such as HTTP message headers from the server and the cache refresh value configured via the command line interface (CLI).

The FreshTime of a file is determined in the following sequence:

1. When a file is downloaded from the HTTP server, if one of the HTTP message headers contains the following:
   
   Cache-Control: max-age = <value in seconds>
   
   Then the max-age is used as the FreshTime for this file.

2. If step 1 does not apply, but the following two headers are included in the HTTP message:
   
   Expires: <expiration date time>
   Date: <Current date time>
   
   Then the difference (Expires – Date) is used as the FreshTime for this file.

3. The HTTP/1.1 spec, RFC 2616 (HyperText Transport Protocol), recommends that either one of the HTTP message headers as described in step 1 or 2 above should be present. If the server fails to send both 1 and 2 in its HTTP response, then take 10% of the difference between Date and Last-Modified from the following message headers:
   
   Last-Modified: <last-modified date time>
   Date: <Current date time>
   
   So the FreshTime for this file is calculated as:
   
   FreshTime = 10% * ((Date) – (Last-Modified))

4. The CLI allows the user to assign a FreshTime value to the files as a provisional value in case none of the message headers in steps 1 to 3 are present:
   
   http client cache refresh <1-864000>
   
   The default refresh value is 86400 seconds (24 hours). The configured HTTP Client cache refresh has no effect on files when any of the message headers in steps 1 to 3 are present. This command is also not retroactive. That is, the newly configured refresh value applies only to new incoming files, and it has no effect on the entries already in the cache.

Stale files are refreshed on an as-needed basis only. This means that a stale cached entry can stay in the cache for a long time until it is removed to make room for either a fresh copy of the same file or another file that needs its memory space in the cache.

A stale cached entry is removed on an as-needed basis when all of the following conditions are true:

- The cached entry becomes stale.
- Its refresh count is zero (0); that is, the cached entry is not being used.
- Its memory space is needed to make room for other entries.

When the Age exceeds the FreshTime and the file needs to be played, the HTTP Client will check with the media server to determine whether or not the file has been updated. When the HTTP Client issues a GET request to the server, it uses a conditional GET to minimize its impact on network traffic. The GET request includes an If-Modified-Since in the headers sent to the server. With this header, the server will either reply with a 304 response code (Not Modified) or return the entire file if the file was indeed updated recently.

Note that this conditional GET applies only to non-streaming mode. Under streaming mode, the HTTP Client always issues an unconditional GET; that is, no If-Modified-Since header is included in the GET request, thus resulting in an unconditional reload for each GET in streaming mode.
You can reload individual files into cache by issuing the following command:

```
test http client get http://10.0.0.130/en-us/sys/1.wav reload
```

## Branch Office Implications

In most cases, customers implementing Unified CVP in branch office deployments expect a small footprint for hardware, and they will not have a local media server. Therefore, it is necessary to store some critical prompts in flash, such as error messages or other messages that are played to the caller when the WAN is down.

When recorded in G.711 mu-law format, typical prompts of average duration are about 10 to 15 kB in size. When sizing gateways for such implementations, size the flash memory by factoring in the number of prompts and their sizes, and also leave room for storing the Cisco IOS image.
CHAPTER 13

Managing, Monitoring, and Reporting

This chapter discusses various types of managing, monitoring, and reporting functions that can be used with Unified CVP. It covers the following areas:

- Operations Console: Managing and Monitoring, page 13-1
- End-to-End Tracking of Individual Calls: Log Files, page 13-2
- Formal Reporting, page 13-3

Operations Console: Managing and Monitoring

The Operations Console has a web-based interface from which you can configure the Unified CVP components in the Unified CVP solution. You can also monitor all components in the Unified CVP solution.

You can manage the following Unified CVP components directly from the Operations Console:

- Call Server
- VoiceXML Server
- Reporting Server

The Operations Console provides web-based interfaces for mapping and summarizing the solution network configuration, setting and displaying configuration information on a batch or per-node basis, and storing local copies of these configurations. The Operations Console also provides the ability to distribute VoiceXML Studio applications to VoiceXML Servers. Finally, the Operations Console provides basic visual indications as to which managed components are functioning properly and which are having problems.

The Operations Console provides access to the following operations:

- Troubleshooting
  You can use the Support Tools site, which provides the ability to retrieve and process trace logs from most components, plus the ability to set or reset trace levels on these components.

- Health Monitoring
  You can use any SNMP-standard monitoring tool to get a detailed visual and tabular representation of the health of the solution network. All Unified CVP product components and most Unified CVP solution components also issue CVP traps and statistics that can be delivered to any standard SNMP management station or monitoring tool.
• Statistical Monitoring
Unified CVP infrastructure statistics include real-time and interval data on the Java Virtual Machine (JVM), threading, and licensing. You can access these statistics by selecting the Control Center from the System menu and then selecting a device. SNMP statistics can also be used.
• Direct administration of individual Cisco IOS-based components
Administrators can select an individual gateway, gatekeeper, or content services switch for direct administration. Secure Shell (SSH) is used for the gateway and gatekeeper, while Telnet is used for the Content Services Switch (CSS).

Note
Internally, the Operations Console is occasionally referred to as the OAMP (Operate, Administer, Maintain, Provision). The Operations Console manages individual components through an OAMP Resource Manager (ORM), which is co-located with each managed Unified CVP component. The ORM is invisible to the end-user.

For more information on the Operations Console, see the Operations Console online help.

End-to-End Tracking of Individual Calls: Log Files

When a call arrives at a Unified CVP ingress gateway, Cisco IOS assigns that call (regardless of whether SIP or H.323 is being used) a 36-digit hexadecimal Global Unique Identifier (GUID) that uniquely identifies the call. Unified CVP carries that GUID through all of the components that the call encounters, as follows:
• Ingress gateway — shown in Cisco IOS log files
• VoiceXML gateway — shown in Cisco IOS log files
• Unified CVP components — shown in Unified CVP log files
• Unified Intelligent Contact Management Enterprise (ICME) — shown in the Extended Call Context (ECC) variable user.task.id and stored with all Termination Call Detail (TCD) and Route Call Detail (RCD) records
• Automatic speech recognition (ASR) and text-to-speech (TTS) servers — shown in logs as the logging tag
• Cisco Unified Communications Manager (Unified CM) — appears in the detailed logs

Thus, with proper levels of logging enabled, a call can be traced through all of the above components. The Unified CVP logs are located in $CVP_HOME/logs. All of the Unified CVP logs roll over at 12:00 AM every night, with the date as part of the filename. The format of the date is yyyy-mm-dd. All of these logs will also roll over when they reach the predefined size limit of 100 MB and will have a number as part of the filename extension. The number indicates which log it was for that day. When the entire logs directory reaches a predefined size, old files are purged as necessary.

For more information on Unified CVP logging, see the Troubleshooting Guide for Cisco Unified Customer Voice Portal, available at


Note
Although Unified CVP components do not themselves synchronize machine times, customers must provide a cross-component time synchronization mechanism, such as NTP, in order to assure accurate time stamps for logging and reporting.
Formal Reporting

The Unified CVP Reporting Server houses the Reporting Service and hosts an IBM Informix Dynamic Server (IDS) database management system.

The Reporting Service provides historical reporting to a distributed self-service deployment in a call center environment. The system is used to assist call center managers with call activity summary information to manage daily operations. It can also provide operational analysis of various IVR applications.

The Reporting Service receives reporting data from the IVR Service, the SIP Service (if used), and the VoiceXML Server. As stated, it is deployed together with an Informix database management system, and it transforms and writes this reporting data into that database. The database schema is prescribed by the Unified CVP product, but the schema is fully published so that customers can develop custom reports based on it.

The Reporting Service does not itself perform database administrative and maintenance activities such as backups or purges. However, Unified CVP provides access to such maintenance tasks through the Operations Console.

There needs to be only one Reporting Server in a deployment. This server does not represent a single point of failure, however, because data safety and security are provided by the database management system, and temporary outages are tolerated due to persistent buffering of information on the source components. However, if more than one Reporting Server is used, be aware of the following restrictions:

- Each Call Server can be associated with only one Reporting Server.
- Reports cannot span multiple Informix databases.

Although Unified CVP components do not themselves synchronize machine times, customers must provide a cross-component time synchronization mechanism, such as NTP, in order to assure accurate time stamps for logging and reporting.

Backup and Restore

Unified CVP utilizes RAID as protection against failure of a single drive in a mirrored pair. However, RAID 10 will not protect against the loss of a site, loss of a machine, or a loss of both mirrored drives.

Unified CVP allows customers, by means of the Operations Console, to schedule daily database backups or to run database backups on-demand. This capability enables the customer to restore the database manually (if needed) to the last backup time, so that the worst-case scenario is losing about 24 hours worth of data.

Database backups are written to the local database server. However, storing backups only on a local machine does not protect the system against server failures or the loss of a site. Cisco recommends that Unified CVP customers copy the backup files to a different machine, preferably at a different location. Customers who choose to do this must assume all security and backup management responsibilities.

Database backups are essentially the same size as the originating database. Due to disk size limitations, Unified CVP can store a maximum of two backups. Customers who wish to store more copies of database backups must copy the backups to another location.

Database restore is not supported through the Operations Console. To restore the Unified CVP database, a customer must manually run the Informix command from a command prompt.
More Information

For more information on Unified CVP reporting, see the Reporting Guide for Cisco Unified Customer Voice Portal, available at

This chapter discusses how to determine how many physical machines to order and, in the case of gateways and gatekeepers, what kind to order.

This chapter covers the following topics:

- Sizing Overview, page 14-1
- Unified CVP Call Server, page 14-2
- Unified CVP VoiceXML Server, page 14-3
- Unified CVP Co-Residency, page 14-3
- Unified Presence Server, page 14-4
- Unified CVP Video Service, page 14-4
- Unified CVP Reporting Server, page 14-7

### Sizing Overview

When sizing a contact center, first determine the worst-case contact center profile in terms of the number of calls that are in each state. In other words, if you were to observe the contact center at its busiest instant in the busiest hour, how many calls would you find are in each of the following states:

- Self-service — Calls that are executing applications using the Unified CVP VoiceXML Server
- Queue and collect — Calls that are in queue for an agent or that 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 talking call is using resources, you must consider how the call gets transferred to that VRU or agent. If the call was transferred via VoIP, it continues to use an ingress 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 re-deliver it at a later time. The same is true of calls that are tromboned 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 should be counted as talking calls.

However, if the call was transferred via *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 should not be counted as talking calls.
Finally, include in the overall call counts those calls that have been transferred back into Unified CVP for queuing or self-service, via either blind or warm methods. For instance, if a warm transfer is used and the agent is queued at Unified CVP during the post-route phase, the call would use two ports due to two separate call control sessions at Unified CVP. Because these calls usually do not amount to more than 5% or 10% of the overall call volume, it is easy to overlook them.

The definitions of these call states differ somewhat from the definitions used for port licensing purposes (see Licensing, page 15-1). The use of automatic speech recognition (ASR) or text-to-speech (TTS) has nothing to do with delineating which calls are in which state, whereas it does for licensing purposes. Similarly, the call state determination has nothing to do with whether the agents are Unified CCE agents or ACD agents, nor does it matter whether the customer intends to use Unified CVP’s ability to retrieve and re-deliver the call to another agent or back into self-service. Note that the solution must be sized 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 will need this information for the contact center as a whole. Because it is hard to identify a true maximum arrival rate, you can use statistical means to arrive at this number. Except in fairly small implementations, this is seldom the critical factor in determining sizing.

With the above data, you can begin sizing each component in the network. This section next considers the Unified CVP products: Call Server and VoiceXML Server followed by the gateways, gatekeepers, and content switches. This section deals entirely 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 Designing Unified CVP for High Availability, page 4-1.

### Unified CVP Call Server

**Note**

The Unified CVP Call Server is not used in Model #1: Standalone Self-Service. This section does not apply to such deployments.

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

Each Unified CVP Call Server can handle 850 SIP calls or 500 H.323 calls. Each Unified CVP Call Server is further limited to a sustained call arrival rate of 7 calls per second (cps). However, Model #4 is exempt from this limitation because the Unified CVP Call Server in that model does not perform any H.323 or SIP processing.

Specifically, the number of Unified CVP Call Servers required is the larger of:

\[
\begin{align*}
\text{((Self Service) + (Queue and Collect) + Talking) / 850 [or 500 for H.323], rounded up} \\
\text{or} \\
\text{(Average call arrival rate) / 7, rounded up [except in Model #4]}
\end{align*}
\]

If you have a Call Server servicing both SIP and H.323 calls, you can size the server by prorating the performance used for each call type. For example, if 60% of the calls will be H.323 and 40% SIP, the maximum load in terms of active calls in this case will be:

\[
60% \times (500 \text{ H.323 calls}) + 40% \times (850 \text{ SIP calls}) = 640 \text{ active calls}
\]
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.

**Unified CVP VoiceXML Server**

Unified CVP VoiceXML Server sizing is simple: one VoiceXML Server can handle up to 750 calls. If you are using Unified CVP VoiceXML Servers, you should size those machines according to the following formula:

\[
\text{Calls} / 750, \text{ rounded up}
\]

where \textit{Calls} refers to the number of calls that are actually in Unified CVP VoiceXML Server self-service applications at that busy moment snapshot in time.

**Unified CVP Co-Residency**

The following components can be installed on the same physical server (co-resident):

- Unified CVP Call Server
- Unified CVP VoiceXML Server
- Media Server

A SIP-based co-resident server can handle 750 SIP calls as well as 750 VoiceXML Server sessions simultaneously, and it can handle a sustained call arrival rate of 6 calls per second. An H.323 co-resident server can handle 500 H.323 calls as well as 500 VoiceXML Server sessions simultaneously, and it can handle a sustained call arrival rate of 6 calls per second.

The number of Unified CVP Call Servers required is the larger of:

\[
\frac{(\text{Self Service}) + (\text{Queue and Collect}) + \text{Talking})}{750} \text{ [or 500 for H.323]}, \text{ rounded up}
\]

or

\[
\frac{\text{Average call arrival rate}}{6}, \text{ rounded up [except in Model #4]}
\]

The co-resident media server can be used for up to 750 calls [or 500 for H.323], assuming that prompt caching is enabled in the VoiceXML gateways. If multiple co-resident servers are to be used, you must load-balance across the co-resident media servers in order to spread the load of the calls across all of the servers. To reduce the administrative overhead of managing content on multiple media servers, separate dedicated media servers can be used.

**Co-Resident Reporting Server and Call Server**

The Unified CVP Reporting Server can also be co-resident with the Call Server, but only for Standalone VoiceXML deployments. The Call Server is normally not needed in a Standalone VoiceXML deployment; but if reporting is desired, a Call Server is required in order to send the reporting data from the VoiceXML Server to the Reporting Server. Thus, when the Unified CVP Reporting server is co-resident with a Call Server, the Call Server is not processing any SIP or H.323 calls but is simply relaying reporting data from the VoiceXML Server.

The co-resident Call Server does not have a significant impact on performance in this model, therefore the sizing information in the section on the Unified CVP Reporting Server, page 14-7, does not change.
Unified Presence Server

The Cisco Unified Presence server is the SIP Proxy Server provided by Cisco for use with Unified CVP. Table 14-1 outlines the performance of the various server types.

Table 14-1  Call Handling Capacities for Cisco Unified Presence Servers

<table>
<thead>
<tr>
<th>Cisco Server Model</th>
<th>Recording Function</th>
<th>UDP</th>
<th>TCP</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCS-7825</td>
<td>Record-Route On</td>
<td>200 cps</td>
<td>100 cps</td>
</tr>
<tr>
<td></td>
<td>Record-Route Off</td>
<td>300 cps</td>
<td>300 cps</td>
</tr>
<tr>
<td>MCS-7835</td>
<td>Record-Route On</td>
<td>200 cps</td>
<td>100 cps</td>
</tr>
<tr>
<td></td>
<td>Record-Route Off</td>
<td>300 cps</td>
<td>300 cps</td>
</tr>
<tr>
<td>MCS-7845</td>
<td>Record-Route On</td>
<td>600 cps</td>
<td>200 cps</td>
</tr>
<tr>
<td></td>
<td>Record-Route Off</td>
<td>1100 cps</td>
<td>500 cps</td>
</tr>
</tbody>
</table>

The sizing numbers in Table 14-1 assume a dedicated SIP Proxy. If you are using the same Cisco Unified Presence server for presence services, you should adjust the numbers accordingly, based on the capacity used. For instance, if the Cisco Unified Presence server supports 2500 presence users but you will actually have only 1250 (50%), then that leaves approximately 50% of the SIP Proxy capacity.

The capacities in Table 14-1 are measured in calls per second (cps). However, one call coming in from the PSTN is not equivalent to one call through Cisco Unified Presence. Multiple calls are actually generated per inbound customer call for queuing, ringback, and subsequent agent transfers. A typical incoming call will be transferred by Unified CVP four times, so the inbound PSTN call rate should be multiplied by 4.

For example:

If Unified CVP receives 20 PSTN calls per second, Cisco Unified Presence will see about 80 calls per second.

Unified CVP Video Service

Cisco Unified CVP release 7.0 introduces capabilities for video self-service and queuing, integrated with video-capable agents of Cisco Unified Contact Center Enterprise (Unified CCE). Unified CVP video capabilities come in two forms: Full Video Service and Basic Video Service.

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 Unified CVP Call Servers must be used for the video and audio calls.
Sizing Unified CVP Basic Video Service

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

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

Sizing Unified CVP Full Video Service

The Unified CVP Full Video Service requires the Unified CVP Call Server plus video-specific components such as Cisco Unified Videoconferencing hardware and third-party software from Radvision, Inc., including the Radvision iContact application and the Radvision Interactive Video Platform (IVP). Additional hardware such as 3G Gateways are also required for 3G mobile video solutions.

CVP Components

Sizing the Unified CVP Call Server for full-service video is done in the same manner as for traditional audio applications. In other words, the Unified CVP Call Server does not care whether a call is an audio call or a video call, therefore video sizing can be done the same way as audio sizing.

The Unified CVP VXML server is not required for video-only deployments because the Unified ICM script editor is the only way to script and run Unified CVP video functionality. However, if a Unified CVP system needs to support both audio and video calls, then the Unified CVP VXML server should be used, and it should be sized in the standard manner for the number of audio calls.

A Unified CVP Video Media Server supports up to 1500 agents connected to the web interface. Of those agents, the number of simultaneous video streams being played to the caller is limited as follows:

- On a Cisco MCS 7845-H2
  - 100% with caching: 850 simultaneous streams
  - 100% without caching: 66 simultaneous streams
- On a Cisco MCS 7825-H3
  - 100% with caching: 425 simultaneous streams
  - 100% without caching: 33 simultaneous streams

In addition, up to 10,000 media files and/or resources can be distributed across up to 5000 directories. Each subfolder counts as a directory.

Cisco Unified Videoconferencing Hardware

The following components are required for the Cisco Unified Videoconferencing hardware that terminates mixes video calls in the solution:

- IPVC-3545-CHAS (the MCU chassis)
- IPVC-3545-MCU (one MCU card required per chassis)
- IPVC-3545-EMP (one EMP card required per chassis, with a maximum of two EMP cards allowed per chassis)
Chapter 14 Sizing

Unified CVP Video Service

The minimum configuration consists of the MCU chassis, the MCU card, and one EMP card (also called an MVP card), and it supports up to 48 H.263 CIF video ports (or up to 96 H.263 QCIF video ports). Adding a second EMP/MVP card to the chassis increases the capacity up to 96 H.263 CIF video ports (H.263 QCIF capacity remains at 96 ports even with the second EMP/MVP card). Note that a deployed video service must be in either the CIF video format (in which QCIF endpoints such as 3G mobile phones are treated like CIF endpoints for MCU sizing purposes) or the QCIF video format. The different formats cannot be mixed as part of the same video service deployment.

Video capacity can be increased beyond this size by adding additional MCU chassis (each with one MCU card and one or two EMP/MVP cards). All MCUs are managed as a pool of media resources by the Radvision IVP controller. When a new call is received, the Radvision IVP can select any of the MCUs it manages to service the call, based on resource availability.

MCU video ports are consumed by all of the following:

- Video caller
- Video agents
- Unified CVP or Radvision playing video to the caller
- Agent playing video to the caller
- Video session being recorded

Video ports are allocated and de-allocated as necessary during the life of a call. The following examples illustrate how many MCU video ports are required in different video call scenarios:

- A caller undergoing video self-service or video queuing consumes two video ports (one for the video caller and one for the video being played to the caller by Unified CVP or Radvision).
- If the above session is recorded, then three video ports are consumed.
- A caller and an agent having a video conversation consume two video ports. If this session is recorded, then three video ports are consumed.
- If an agent plays a video to the caller, then three video ports are consumed. If this session is recorded, then four video ports are consumed.
- If video agent #1 performs a blind transfer to video agent #2, two video ports are consumed while the caller is on hold (one for the caller and one for the on-hold video being played to the caller). Once agent #2 becomes available, two video ports are still consumed (one for the caller and one for agent #2). If agent #2 plays a video to the caller, or if the session is recorded, then additional video ports are consumed, as described above.
- If video agent #1 performs a consultative transfer (by bringing on agent #2 while agent #1 stays on the call), three video ports are consumed (one for the caller, and one for each of the two video agents).

Other Software and Hardware

Sizing information for the Radvision IVP, iContact, and 3G Gateway products should be obtained from Radvision.

VXML Gateways are not required if the solution needs to support Full Service Video only. However, if a Unified CVP system needs to support both audio and video calls, then VXML Gateways are required for the audio calls, and they should be sized in the standard manner.
Unified CVP Reporting Server

There are many variables to take into account when sizing the Unified CVP Reporting Server. Different VoiceXML applications have different characteristics, and those characteristics play a large part in 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 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 will be generated by your VoiceXML application. The example applications and the tables in subsequent sections of this chapter will 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 number of minutes customers will spend receiving VoiceXML call treatment by that application.
2. Estimate the calls per second that the application will receive.
3. 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:

\[ A# = \%CPS \times CPS \times MSG / Min / 60 \]

Where:

- \( A# \) = the number of estimated reporting messages per second for an application. Complete one calculation per application (A1, A2, …, An).
- CPS = the number of calls per second.
- \( \%CPS \) = the percentage of calls that use this VoiceXML application.
- MSG = 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, page 14-8, and Example Applications, page 14-9.
- Min = Amount of time spent in the application (in minutes).
- 60 = the number of seconds in one minute.

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

\[ A(\text{total}) = A1 + A2 + \ldots + An \]

This is the total number of reporting messages generated per second by your VoiceXML applications. The Cisco MCS-7845 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 it is greater, you need to use multiple reporting servers and partition the VoiceXML applications to use specific reporting servers.
How to Use Multiple Reporting Servers

If the number of messages per second (as determined in steps 1 and 2 above) 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 each VoiceXML Server can be associated with only one Reporting Server.
- Reports cannot span multiple Informix databases.


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 non-interesting data creates more usable data repositories that support higher message volume.
- Configure the dial plan and/or other available means to direct the incoming calls to the appropriate Call Server and VoiceXML Server.

If you need to combine data from multiple databases, possible options may include:

- 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

Table 14-2 outlines the various elements or activities and the number of reporting messages generated by each.

<table>
<thead>
<tr>
<th>Element or Activity</th>
<th>Number of Reporting Messages (Unfiltered)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start$^1$</td>
<td>2</td>
</tr>
<tr>
<td>End$^1$</td>
<td>2</td>
</tr>
<tr>
<td>Subdialog_start$^1$</td>
<td>2</td>
</tr>
<tr>
<td>Subdialog_return$^1$</td>
<td>2</td>
</tr>
<tr>
<td>Hotlink</td>
<td>2</td>
</tr>
<tr>
<td>HotEvent</td>
<td>2</td>
</tr>
<tr>
<td>Transfer w/o Audio</td>
<td>2</td>
</tr>
<tr>
<td>Currency w/o Audio</td>
<td>2</td>
</tr>
</tbody>
</table>
Table 14-2  Number of Reporting Messages per Element or Activity (continued)

<table>
<thead>
<tr>
<th>Element or Activity</th>
<th>Number of Reporting Messages (Unfiltered)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flag</td>
<td>2</td>
</tr>
<tr>
<td>Action</td>
<td>2</td>
</tr>
<tr>
<td>Decision</td>
<td>2</td>
</tr>
<tr>
<td>Application Transfer</td>
<td>2</td>
</tr>
<tr>
<td>VXML Error</td>
<td>2</td>
</tr>
<tr>
<td>CallICMInfo (per call)</td>
<td>2</td>
</tr>
<tr>
<td>Session Variable (per change)</td>
<td>2</td>
</tr>
<tr>
<td>Custom Log (per item)</td>
<td>2</td>
</tr>
<tr>
<td>Play (Audio file or TTS)</td>
<td>2</td>
</tr>
<tr>
<td>Get Input (DTMF)</td>
<td>5</td>
</tr>
<tr>
<td>Get Input (ASR)</td>
<td>9</td>
</tr>
<tr>
<td>Form</td>
<td>10</td>
</tr>
<tr>
<td>Digit_with_confirm</td>
<td>20</td>
</tr>
<tr>
<td>Currency_with_confirm</td>
<td>20</td>
</tr>
<tr>
<td>ReqICMLabel</td>
<td>30</td>
</tr>
</tbody>
</table>

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

Example Applications

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

Low Complexity
Total: 16 reporting messages per minute per call.

<table>
<thead>
<tr>
<th>Element Type</th>
<th>Approximate Number of Reporting Messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>2</td>
</tr>
<tr>
<td>Subdialog_start</td>
<td>2</td>
</tr>
<tr>
<td>Play element</td>
<td>2</td>
</tr>
<tr>
<td>Play element</td>
<td>2</td>
</tr>
<tr>
<td>Play element</td>
<td>2</td>
</tr>
<tr>
<td>Play element</td>
<td>2</td>
</tr>
<tr>
<td>Play element</td>
<td>2</td>
</tr>
<tr>
<td>Subdialog_end</td>
<td>2</td>
</tr>
<tr>
<td>End</td>
<td>2</td>
</tr>
</tbody>
</table>
Medium Complexity DTMF Only
Total: 39 reporting messages per minute per call.

<table>
<thead>
<tr>
<th>Element Type</th>
<th>Approximate Number of Reporting Messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>2</td>
</tr>
<tr>
<td>Subdialog_start</td>
<td>2</td>
</tr>
<tr>
<td>Play element</td>
<td>2</td>
</tr>
<tr>
<td>Get input</td>
<td>5</td>
</tr>
<tr>
<td>Play element</td>
<td>2</td>
</tr>
<tr>
<td>Get input</td>
<td>5</td>
</tr>
<tr>
<td>Form</td>
<td>10</td>
</tr>
<tr>
<td>Input</td>
<td>5</td>
</tr>
<tr>
<td>Transfer with audio</td>
<td>2</td>
</tr>
<tr>
<td>Subdialog_end</td>
<td>2</td>
</tr>
<tr>
<td>End</td>
<td>2</td>
</tr>
</tbody>
</table>

Medium Complexity Using Automatic Speech Recognition (ASR)
Total: 51 reporting messages per minute per call.

<table>
<thead>
<tr>
<th>Element Type</th>
<th>Approximate Number of Reporting Messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>2</td>
</tr>
<tr>
<td>Subdialog_start</td>
<td>2</td>
</tr>
<tr>
<td>Play element</td>
<td>2</td>
</tr>
<tr>
<td>Get input</td>
<td>9</td>
</tr>
<tr>
<td>Play element</td>
<td>2</td>
</tr>
<tr>
<td>Get input</td>
<td>9</td>
</tr>
<tr>
<td>Form</td>
<td>10</td>
</tr>
<tr>
<td>Input</td>
<td>9</td>
</tr>
<tr>
<td>Transfer with audio</td>
<td>2</td>
</tr>
<tr>
<td>Subdialog_end</td>
<td>2</td>
</tr>
<tr>
<td>End</td>
<td>2</td>
</tr>
</tbody>
</table>
High Complexity Using Automatic Speech Recognition (ASR)

Total: 107 reporting messages per minute per call.

<table>
<thead>
<tr>
<th>Element Type</th>
<th>Approximate Number of Reporting Messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>2</td>
</tr>
<tr>
<td>Subdialog_start</td>
<td>2</td>
</tr>
<tr>
<td>Icmrequestlabel</td>
<td>30</td>
</tr>
<tr>
<td>Form</td>
<td>10</td>
</tr>
<tr>
<td>ASR capture</td>
<td>9</td>
</tr>
<tr>
<td>Digit with confirm</td>
<td>20</td>
</tr>
<tr>
<td>Form</td>
<td>10</td>
</tr>
<tr>
<td>Digit with confirm</td>
<td>20</td>
</tr>
<tr>
<td>Subdialog_end</td>
<td>2</td>
</tr>
<tr>
<td>End</td>
<td>2</td>
</tr>
</tbody>
</table>
Licensing

The licensing information for Cisco Unified CVP has been consolidated and moved into the Cisco Customer Contact Solutions Ordering Guide. The Ordering Guide provides a single, frequently updated source for all the Unified CVP licensing information. Cisco employees and Partners with a valid login account can access the Ordering Guide at:


If you need licensing information for Unified CVP but you cannot access the Ordering Guide, contact your local Cisco Systems Engineer (SE) or Partner.
## Symbols

*8 TNT  3-4, 10-2, 14-1  
?  12-4  

### A

ACD  5-13  
admission control for calls  3-4  
aging cache  12-4  
alternate  
  endpoints  4-22  
  gatekeeper  4-12, 4-13  
application examples  14-9  
arbor architecture  1-1  
ASR  1-12, 4-27, 7-2, 9-6  
automatic call distributor (ACD)  5-13  
Automatic Speech Recognition (ASR)  1-12, 4-27, 7-2, 9-6  

### B

backup and restore  13-3  
bandwidth  
  for retrieving prompts  12-3  
  provisioning  9-1, 9-4  
Basic Video Service  2-13, 14-5  
blind transfer  6-2  
blocking media streams  9-9  
border element  7-3  
branch office  
  gateways  3-1  
  media server  12-6  

### C

cache aging  12-4  
caching  
  prompts  12-3  
  query URLs  12-4  
call admission control  9-8  
Call Director  
  deployment model  2-3, 5-7, 6-3  
calls  
  admission control  3-4  
  control of  2-2, 2-5, 2-10  
  control traffic  9-2  
  disposition of  4-5, 4-8, 4-10, 4-13, 4-16, 4-17, 4-22, 4-23, 4-25, 4-26, 4-27, 4-28, 4-29, 4-30  
  failures  4-13  
  flows  1-13, 2-2, 2-4, 2-5, 2-7, 2-8, 2-11, 2-15, 3-5, 3-8, 6-2, 6-3, 11-3  
  help desk  6-2  
  initial treatment  2-7, 2-8, 2-11  
  in progress  4-9, 4-15  
  log files  13-2  
  maximum number  7-7, 9-7  
  originated by Cisco Unified CM  5-13, 6-1  
  outbound  6-2  
  post-ICM  11-4, 11-5  
  pre-routing  11-3, 11-4, 11-5  
  queue and collect  14-1  
  routing  3-9, 5-11  
  self-service  14-1  
  survivability  3-4  
  tracking  13-2  
  traffic  9-2  
  transfers  2-2, 2-4, 2-5, 2-7, 2-9, 2-10, 2-12, 10-1
Call Server 14-2
CCE 1-9
Central Controller 9-3
centralized
   VoiceXML gateways 4-18
   VoiceXML servers 3-3
Cisco IOS 4-6, 4-16, 12-3
Cisco Unified Border Element 7-3
Cisco Unified Presence 14-4
Cisco Unified Videoconferencing 14-5
clusters 3-7
co-located VoiceXML servers and gateways 3-2
components of CVP 1-3
Comprehensive deployment model
described 1-2
   Using CVP VoiceXML Server 5-7, 6-6
   Using ICM Micro-Apps 5-7, 6-5
configuration of
   ASR 4-27
   caching for prompts 12-3
Cisco IOS 4-16
Cisco IOS gateway 4-6
Cisco Unified CM 4-29, 6-7
Content Services Switch (CSS) 4-23
dial plan 6-7
gatekeeper 4-12, 4-13
H.323 4-14
HSRP 4-12
Intelligent Contact Management (ICM) 4-30
IVR service 4-17
media server 4-24
originating gateway 4-4
SIP Proxy Server 4-6, 4-9
streaming for prompts 12-3
TTS 4-27
Unified ICM 6-6
VoiceXML gateway 4-18, 4-19
VoiceXML Server 4-26
consultative transfer 6-2

Content Services Switch (CSS) 1-11, 4-23
control traffic 9-2
co-resident
   ingress gateway and VoiceXML servers 4-19
Correlation ID 5-2, 5-4
CSS 1-11, 4-23
CVP
   architecture 1-1
   Call Server 14-2
   components 1-3
   co-residency 14-3
described 1-2
   GKTMP 11-2
   licensing 15-1
   Operations Console 13-1
   Server 1-4
   sizing components 14-1
   Studio 8-3
   Video Media Server 1-6
   Video Service 4-25, 14-4

data
   reporting 13-1
   traffic 9-4
deployment models
   Call Director 2-3
   Comprehensive models 2-6
distributed models 3-1
   functional models 1-15, 2-1
   hosted implementations 5-9, 6-7
   Model #1 - Standalone Self-Service 5-7, 6-3
   Model #2 - Call Director 5-7, 6-3
   Model #3a - Comprehensive Using ICM Micro-Apps 5-7, 6-5
   Model #3b - Comprehensive Using CVP VoiceXML Server 5-7, 6-6
Model #4a - VRU Only with NIC Controlled Routing 5-8
Model #4b - VRU Only with NIC Controlled Pre-Routing 5-8
Model #4 - VRU Only 5-8
Network VRU types 5-5
standalone self-service 4-26, 4-28
Standalone VoiceXML Server 2-1
VRU only 2-10
design process 1-14
dial peers 3-9
dial plan 6-7
disposition of calls 4-5, 4-8, 4-10, 4-13, 4-16, 4-17, 4-22, 4-23, 4-25, 4-26, 4-27, 4-28, 4-29, 4-30
distributed deployments 3-1
gateways 3-1
network options 1-15
VoiceXML gateways 4-19
DNS Server 1-10
DTMF 7-2

E
Egress Gateway 1-7
example applications 14-9

F
firewalls 9-10
flow of calls 6-2, 6-3, 11-3
formal reporting 13-3
Full Video Service 2-13, 14-5
functional deployment models 1-15, 2-1

G
G.711 9-9
gatekeeper
alternate 4-12, 4-13
call admission control 3-5
call routing 3-9
call routing configuration 4-12, 6-7
described 1-9
H.323 3-9
high availability 4-11
HSRP 4-11, 4-12
redundancy 4-11
Gatekeeper Transaction Message Protocol (GKTMP) 11-1
gateways
at a branch office 3-1
centralized 4-18
Cisco IOS 4-6, 4-16
co-located with VoiceXML servers 3-2
distributed 3-1, 4-19
maximum number of calls 7-7
maximum VoiceXML sessions 7-6
MGCP 7-7
originating calls 4-4
PSTN 7-1, 7-2
selecting appropriate ones 7-1, 7-4
sizing 6-8, 7-5
using Cisco Unified CM 3-3
voice egress 1-7
voice ingress 1-6
VoiceXML 1-7, 3-1, 4-17, 7-2
GED-125 9-3
GKTMP 11-1

H
H.323
call flow 1-14, 2-5, 2-8, 3-5
characteristics 1-14
call routing configuration 4-14
gatekeeper 3-9
Refer transfer 10-5
Index

IN-4
Cisco Unified Customer Voice Portal (CVP) 7.x Solution Reference Network Design (SRND)

Service 4-14
signaling 9-6
hardware for high availability 4-22
health monitoring 13-1
help desk calls 6-2
high availability
  deployment options 1-16
  design considerations 4-1
  Layer 2 switch 4-3
hookflash 3-4, 10-2, 14-1
hosted implementations 5-9, 6-7
HSRP
  configuration 4-12
  gatekeeper redundancy 4-11
HTTP 8-1

L
Layer 2 switch 4-3
licensing 15-1
log files 13-2

M
managing the Unified CVP system 13-1
maximum
  number of calls 7-7, 9-7
  VoiceXML sessions 7-6
MCU 1-7
media files 9-5, 12-1
Media Gateway Control Protocol (MGCP) 7-7
Media Resource Control Protocol (MRCP) 9-3
media server 1-12, 4-24
media termination point (MTP) 6-8
messages for reporting 14-8
MGCP 7-7, 9-3
microapplications 4-24
Model #1 - Standalone Self-Service 5-7, 6-3
Model #2 - Call Director 5-7, 6-3
Model #3a - Comprehensive Using ICM Micro-Apps 5-7, 6-5
Model #3b - Comprehensive Using CVP VoiceXML Server 5-7, 6-6
Model #4a - VRU Only with NIC Controlled Routing 5-8
Model #4b - VRU Only with NIC Controlled Pre-Routing 5-8
Model #4 - VRU Only 5-8
monitoring the Unified CVP system 13-1
MRCP 9-7
MTP 6-8
multi-language support 8-2
multiple reporting servers 14-8
Multipoint Conference Unit (MCU) 1-7

IBM Informix Dynamic Server (IDS) 13-3
ICM
call transfers 10-4
  Central Controller 9-3
  configuration 6-6
  high availability 4-30
  interactions with CVP 5-1
  with AST/TTS 4-28
  with VoiceXML Server 4-26
IDS 13-3
IN 10-5
Informix Dynamic Server (IDS) 13-3
infrastructure of the network 9-1
Ingress Voice Gateway 1-6
initial call treatment 2-7, 2-8, 2-11
in-progress calls 4-9, 4-15
Intelligent Contact Management (ICM) 4-30, 5-1
Intelligent Network (IN) Release Trunk Transfers 10-5
IOS 4-6, 4-16, 12-3
IVR Service 4-16
Index

**N**

- network infrastructure 9-1
- Network Interface Controller (NIC) 11-1
- network security 9-10
- Network VRU types 5-1, 5-5, 5-12
- NIC 11-1
- non-streaming prompts 12-3

**O**

- OAMP 13-2
- OAMP Resource Manager (ORM) 13-2
- Operate, Administer, Maintain, Provision (OAMP) 13-2
- Operations Console 13-1
- Operations Console Server 1-6
- originating gateway 4-4
- ORM 13-2
- outbound calls 6-2

**P**

- peripheral gateway (PG) 9-3
- PG 9-3
- ports
  - usage 9-9, 9-10
- post-ICM calls 11-4, 11-5
- pre-routing 11-3, 11-4, 11-5
- presence 14-4
- Presence Server 1-9
- prompts
  - bandwidth 12-3
  - caching 12-3
  - non-streaming 12-3
  - streaming 12-3
- protocol-level call flow 2-2, 2-11, 2-15
- provisioning bandwidth 9-1, 9-4
- PSTN gateways 7-1, 7-2

**Q**

- QoS 9-1, 9-9
- Quality of Service (QoS) 9-1, 9-9
- query URLs 12-4
- queue-and-collect calls 14-1

**R**

- Radvision components 1-11, 4-27, 14-5
- RAID 13-3
- RAS 11-1
- redundant
  - gatekeepers 4-11
- Refer transfer 10-5
- Registration Admission Status (RAS) 11-1
- release trunk transfers 10-1
- reporting
  - described 13-1
  - examples 14-9
  - messages 14-8
  - multiple servers 14-8
  - Server 1-5
  - servers 14-7
- Resource Reservation Protocol (RSVP) 3-9, 9-8
- restoring data files 13-3
- routing calls 5-11
- RSVP 3-9, 9-8

**S**

- SBC 7-3
- scalability options 1-16
- scripting 4-25
- security
  - on the network 9-10
- self-service
  - calls 2-7, 2-8, 2-11, 14-1
  - deployment model 4-26, 4-28
separate ingress gateway and VoiceXML servers
   Cisco Unified Presence co-resident
   multiple reporting sizing
   VoiceXML
session border controller (SBC)
SIP
call flow call transfers characteristics
dial plan
Proxy Server
signaling
SIP Service
sizing
   components scalability options
skill groups
Standalone Self-Service deployment model
statistical monitoring
streaming of prompts
Studio
   server
survivability of calls

T
Takeback-and-Transfer (TNT)
TBCT
TCP socket persistence
TDM interface
Telecom Italia Mobile (TIM)
Text-to-Speech (TTS)
third-party
media server

 VRUs
TIM
TNT
traffic
   marking voice
transfers
   blind
call transfer options consultative
   in Call Director deployments in Comprehensive deployments in standalone VoiceXML deployments to live agent VoIP-based warm
Translation Route ID
troubleshooting
TTS
Two B Channel Transfer (TBCT)
Type 10 VRU
Type 2 VRU
Type 3 VRU
Type 5 VRU
Type 7 VRU
Type 8 VRU
types of Network VRUs

U
Unified CM
   as egress gateway
   as ingress gateway
call admission control
calls originated by configuration
described
   high availability
   multiple clusters
Unified Contact Center Enterprise (CCE) 1-9
Unified Presence 14-4
Unified Presence Server 1-9

V

videoconferencing 14-5
video endpoints 1-8
Video Media Server 1-6
Video Service 4-25, 14-4
VMS 1-6
voice response unit (VRU) 5-14
voice traffic 9-2, 9-8
VoiceXML
  alternate endpoints 4-22
  call transfers 10-6
  centralized servers 3-3
  described 1-1
  documents 9-4
Gateway 1-7
gateways 3-1, 4-17, 7-2
maximum number of sessions 7-6
over HTTP 8-1
Server 1-4, 4-25, 8-1, 14-3
sizing 14-3
standalone server 2-1
Studio 1-5, 4-25
VoIP-based
  pre-routing 2-4, 2-5
  transfers 2-4, 2-5
VRU 5-14
VRU Only deployment model 2-10, 5-8
VRU Only with NIC Controlled Pre-Routing 5-8
VRU Only with NIC Controlled Routing 5-8
VRU PG 9-3

W

warm consultative transfer 6-2
Web application servers 8-2
wink 10-2