IP COMMUNICATIONS

Cisco Unified Customer Voice Portal

Building Unified Contact Centers

ciscopress.com

Rue Green

Cisco Unified Customer Voice Portal

Building Unified Contact Centers

Rue Green

Cisco Press

800 East 96th Street

Indianapolis, IN 46240

Cisco Unified Customer Voice Portal

Building Unified Contact Centers

Rue Green

Published by:
Cisco Press
800 East 96th Street
Indianapolis, IN 46240 USA

First Printing December 2011

Library of Congress Cataloging-in-Publication data is on file.

ISBN-13: 978-1-58714-290-1

ISBN-10: 1-58714-290-2

Warning and Disclaimer

This book is designed to provide information about Cisco Unified Customer Voice Portal. Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

The information is provided on an "as is" basis. The author, Cisco Press, and Cisco Systems, Inc., shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the discs or programs that may accompany it.

The opinions expressed in this book belong to the author and are not necessarily those of Cisco Systems, Inc.

Trademark Acknowledgments

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Cisco Press or Cisco Systems, Inc. cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Corporate and Government Sales

The publisher offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales, which may include electronic versions and/or custom covers and content particular to your business, training goals, marketing focus, and branding interests . For more information, please contact: **U.S. Corporate and Government Sales** 1-800-382-3419 corpsales@pearsontechgroup.com

For sales outside of the U.S. please contact: **International Sales international@pearsoned.com**

Feedback Information

At Cisco Press, our goal is to create in-depth technical books of the highest quality and value. Each book is crafted with care and precision, undergoing rigorous development that involves the unique expertise of members from the professional technical community.

Readers' feedback is a natural continuation of this process. If you have any comments regarding how we could improve the quality of this book, or otherwise alter it to better suit your needs, you can contact us through e-mail at feedback@ciscopress.com. Please make sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.

Publisher: Paul Boger

Associate Publisher: Dave Dusthimer

Executive Editor: Brett Bartow

Managing Editor: Sandra Schroeder

Project Editor: Seth Kerney

Editorial Assistant: Vanessa Evans

Cover Designer: Sandra Schroeder

Composition: Mark Shirar

Business Operation Manager, Cisco Press: Anand Sundaram

Manager Global Certification: Erik Ullanderson

Development Editor: Marianne Bartow

Copy Editor: Apostrophe Editing Services

Technical Editors: Gary Ford, Jeff Spronk,
Rahul Manikitala,and Lou Yao

Proofreader: Language Logistics LLC, Christal White

Indexer: Tim Wright

About the Author

Rue Green, CCIE No. 9269, is a Technical Leader for the Customer Collaboration Service Line within Cisco Advanced Services focusing on Unified Contact Center architectures and deployment methodologies. He currently acts in a delivery architect role over sighting deployment architectures for Unified CVP, Unified ICM, and Cisco Unified Communications Manager for Unified Contact Center Solutions. Rue has spent the last 21 years working within different roles covering software development to network architecture, design, and implementation for large voice and data networks. Rue has a B.S. degree in computer science and mathematics from Colorado Mesa University. He is also a dual Cisco Certified Internetwork Expert (CCIE #9269) in Routing & Switching and Voice, and a Certified Information System Security Professional (CISSP #75393). He also carries many other vendor and industry certifications such as the Microsoft Certified Systems Engineer with Exchange 2003 emphasis and the newer Microsoft Certified IT Professional: Enterprise Administrator certification on Server 2008.

About the Technical Reviewers

For more than 13 years, **Gary Ford** has been privileged to work for many large systems integration companies, Cisco Advanced Technology Partners, and end customers, designing, deploying, and maintaining Cisco telephony and contact center solutions. His introductory role to contact centers started in 1997 while working for British Telecom (BT) as a test engineer tasked with integrating the GeoTel ICR platform into BT's core telephony network. Over the following years, Cisco acquired GeoTel and rapidly transformed the ICR product set to include solutions from other Cisco acquisitions and a great deal of in-house innovation. His role has changed over the years from test engineer to contact center and unified communications consultant. Gary spends much of his time designing and deploying Cisco unified communications solutions for a wide range of customers. Gary also holds a Bachelor's of Engineering degree in computer systems engineering, the status of Chartered Engineer, and several Cisco, Microsoft, and business-related professional qualifications.

Jeff Spronk, CCIE No. 20437, has been working in Advanced Services at Cisco as a Network Consulting Engineer for three years. As a member of the Customer Collaboration Service Line, his focus is to design, deploy, and test Cisco Unified Contact Center solutions for large enterprise customers. Jeff's role also includes the creation of UCC-related intellectual property and documentation. Prior to joining Cisco, Jeff worked as part of the UCC deployment team, assisting a Cisco Partner and Cisco sales teams with designing and troubleshooting contact center solutions.

Dedications

This book is dedicated first to my family, my wife Marcy and my two awesome children, Kyla and Jake. Without their support, encouragement, and patience, it would not exist. Secondly to my Mom and Dad, who instilled in me strong work ethics and a will to never doubt myself. Lastly, to my sister, who has always been the big sister with a big heart. I think they will be proud of what follows.

Acknowledgments

I would like to thank the following teams for helping me create this book.

First, my manager and colleague, Aaron Chaskelis from Cisco, for his support and guidance on this project. I could not have done it without it.

Secondly, Janet Byron, formerly a principal engineer at Cisco, for her deep technical knowledge on Unified CVP and guidance on content found throughout the book. I want to wish Janet well in her newly found freedom and hope to have the pleasure of working with her again in the future.

Scott Hogg, a principal consultant at GTRI, for convincing me to step up and take a swing at the book. Without his words of encouragement and guidance, this book would not exist.

Chris Chandler and Tom Armstrong from Cisco for their contributions pertaining to sizing and mitigation techniques for high-latency networks and Unified CVP. Their work, illustrated in this book, is a superb example of their deep technical expertise in solution architecture and Unified CVP.

Jason Kuo, formerly an engineer at Cisco , for his contribution around media files and Unified CVP's IVR and HTTP caching techniques. His content was a great addition to this book, and I am grateful for his insight.

Rahul Manikitala, a Systems Engineer at Cisco Systems, Inc., for his great insight pertaining to the layout and content of this book.

Lou Yao, Network Consulting Engineer at Cisco, for his friendship and honest opinions pertaining to the technical content found in this book.

Andrew Marc-Aurele, Network Consulting Engineer at Cisco, for gathering some great examples pertaining to troubleshooting TDM and VXML conversations.

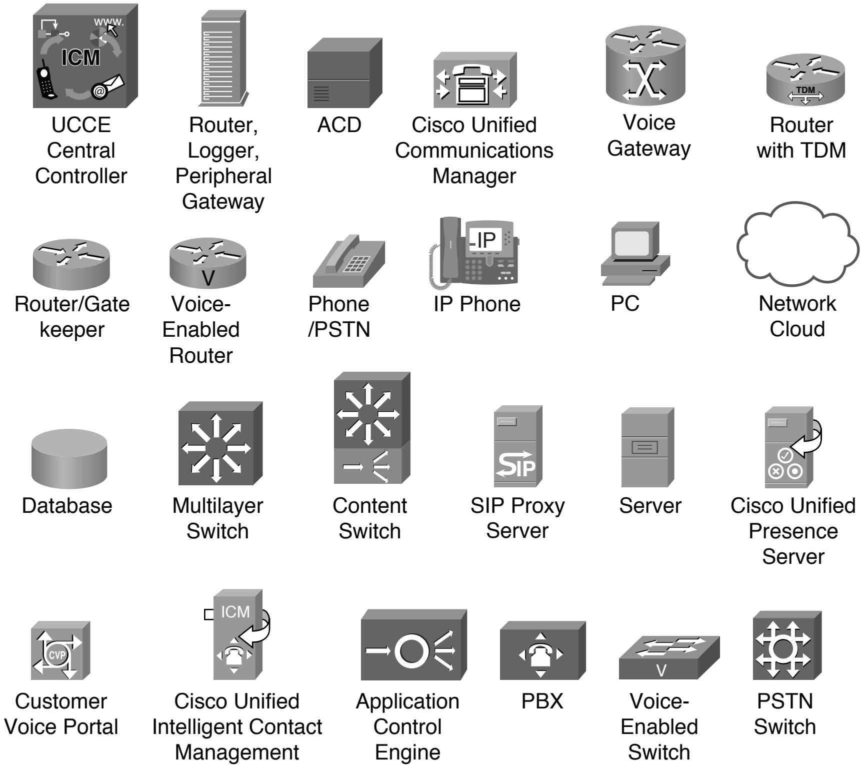
The technical reviewers, Gary Ford and Jeff Spronk, who provided excellent technical coverage and kept this book accurate and easy to navigate.

Finally, the Cisco Press team: Brett Bartow, the executive editor, for seeing the value and vision provided in the original proposal and believing enough to provide me the opportunity to build this book. In addition, Marianne Bartow, development editor, for her relentless push to develop my rough manuscript into a fine piece of technical literature and pushing the entire team to meet our deadlines. Lastly, everyone else in the Cisco Press team who spent countless hours normalizing the manuscript, its technical drawings and content; their effort can be seen throughout the book pertaining to my ideas, words, and pictures, presented in ways that I could never have imagined.

Contents at a Glance

Contents

Icons Used in This Book



Command Syntax Conventions

The conventions used to present command syntax in this book are the same conventions used in the IOS Command Reference. The Command Reference describes these conventions as follows:

- **Boldface** indicates commands and keywords that are entered literally as shown. In actual configuration examples and output (not general command syntax), boldface indicates commands that are manually input by the user (such as a **show** command).
- *Italic* indicates arguments for which you supply actual values.
- Vertical bars (|) separate alternative, mutually exclusive elements.
- Square brackets ([ ]) indicate an optional element.
- Braces ({ }) indicate a required choice.
- Braces within brackets ([{ }]) indicate a required choice within an optional element.

Introduction

Once upon a time I was told that Unified CVP is both a product and a solution. At the time, I didn't understand the significance of that statement and chalked it up as another one of those fancy analogies that actually didn't mean much. However, as time went by I started to examine that statement and understand how correct it truly is. To understand and appreciate Unified CVP, you must first understand it as a product and then graduate to understanding it as a solution. In other words, Unified CVP requires a solution-level mind set. This means to master Unified CVP and it integrations, you must visualize it at the solution level and not just at the product level.

Cisco provides an enormous amount of documentation for Unified CVP. This documentation covers the design, installation, configuration, and administration aspects of Unified CVP. So why write a book on Unified CVP? The answer is fairly simple. With all the great documentation available from Cisco, not a single document provides the important details of Unified CVP. Those important details are scattered throughout numerous Cisco technical documents or buried in training manuals provided by training partners. Up until the construction of this book, an engineer would need to attend some classes and dig through several hundred if not thousands of pages of documentation to just get a good architecture understanding of Unified CVP. This approach created a perception that Unified CVP is a complicated product with a decent learning curve for an engineer to become proficient at implementing it. Hence, the goal of this book is to boil down and simplify the architectural details and collect and present them in one reference without trying to replace the existing design, installation, and configuration guides already available from Cisco.

Currently, Cisco has released version 8.5 of Unified CVP and has a release date for version 8.7 and even 9.0. However, quite a few customers may still use version 7.x or even 4.x, so there is a compelling need for a book that can capture the best technical content sampled from different Cisco documents, white papers, best practices, and brain dumps from senior engineering resources such as the author to present it in a simplified manner for ease of consumption.

Objectives of This Book

Many organizations are currently working to transform their existing legacy TDM Contact Centers to feature-rich, IP-based unified contact centers. Although some organizations are working on these huge transformational projects, several are still contemplating how this transformation can impact them and what solutions exist to enable them to move to a true IP-based unified contact center solution.

Cisco Unified Customer Voice Portal (CVP), formally Internet Service Node or ISN, is quickly becoming a strong replacement technology for legacy TDM IVR solutions. Cisco Unified CVP integrates with Cisco Unified Contact Center Enterprise and Cisco Unified Call Manager to produce a feature-rich replacement for legacy components ailing from scalability, functionality, and proprietary IVR languages. Because of the numerous options and integrations available for designing next generation contact center solutions with Unified CVP, outlining best practices and architectures pertaining to these options becomes an even greater challenge for contact center and network engineers. This book intends to provide architecture guidelines and proven deployment best practices for mitigating design and sizing challenges when deploying Unified CVP.

This book covers the Unified CVP architecture first, outlining its key advantages and design considerations. When the underlining architecture and integration points are defined, the book focuses on key architecture and solutions to address some the most common, yet complex, design challenges existing in today's unified contact center deployments. Where appropriate, the book provides working configurations and examples that support the deployment of the architectures discussed. The book concludes by covering topics such as upgrades, troubleshooting, and the virtualization of Unified CVP.

This book does not cover introductory concepts on how a contact center functions. It is based on the assumption that you have a fundamental understanding of contact centers and their requirements both from a technical standpoint and from a business perspective. In addition, although the audience level for this book may not be expert on Unified CVP, it is assumed that basic unified contact center components such as UCCE via Unified ICM and CUCM are understood. It covers the design architectures for Unified CVP but at the same time gives practical examples on how to implement those architectures. In that way this book provides a good mix of the design concepts with implementation examples for Unified CVP.

Who Should Read This Book?

The book is targeted to a technical audience composed of information technology staff responsible for designing and deploying Unified Contact Centers. In addition, Contact Center mangers that are curious of the design benefits and considerations with Unified CVP could also be likely candidates. Application consultants that provide UCCE scripting support can also benefit from this book, simply because it offers the underlining architecture for how applications are handled in the network and how calls are delivered for queuing, treatment, and delivery. This book assumes some knowledge of IP-based or legacy-based contact centers and is geared to a technical audience. Therefore, people with either a CCVP or CCIE can value the technical content of this book.

The secondary target is the actual end customers in charge of day-to-day maintenance and troubleshooting of their own platform.

How This Book Is Organized

This book contains nine chapters that cover the core areas of Unified Customer Voice Portal. An overview of each chapter follows.

- **Chapter 1, "Introduction to Unified Customer Voice Portal":** Provides a history lesson about Unified CVP and its advantages. This chapter also provides a technical overview of VoiceXML.

- **Chapter 2, "Unified CVP Architecture Overview":** Covers Unified CVP native components including the Call Server, VXML Server, Reporting Server, Operations Console Server, and Cisco Unified Call Studio. This chapter also covers non-native components including IOS devices, Cisco Unified ICM, Cisco Unified Call Manager, content load balancers and third-party servers. This chapter discusses different deployment models and licensing requirements.

- **Chapter 3, "Functional Deployment Models and Call Flows":** Includes a discussion regarding the Standalone, Call Director, Comprehensive, and VRU-only deployment models and their detailed call flows.

- **Chapter 4, "Designing Unified CVP for High Availability":** Covers different Unified CVP geographic models, edge queuing techniques, and call survivability. Includes detailed discussions about the creation of high-availability architectures for SIP, H.323, and content present by load balancers and media servers.

- **Chapter 5, "Working with Media Files":** Covers the architecture of the IOS-based IVR and HTTP Client with discussions pertaining to streaming, caching, and various types of HTTP connections and interactions.

- **Chapter 6, "Sizing, Networking, and Security Considerations":** Covers Sizing, Quality of Service, Network Latency, and Security considerations for Unified CVP.

- **Chapter 7, "Upgrading":** Provides guidance for Unified CVP upgrade strategies and methodologies including approaches to migration H.323 deployments to SIP.
- **Chapter 8, "Troubleshooting":** Provides a framework that you can use to isolate faults with a Unified CVP deployment, followed by continued discussion about how to determine device status and detailed troubleshooting steps for native and non-native components.
- **Chapter 9, "Virtualization":** Covers the history of virtualization and how it applies to Unified CVP deployments, followed by some best practice guidance for designing a virtualized Unified CVP deployment using UCS. This chapter concludes with some use cases for deployments using Unified ICM, Unified CVP, with Unified CCE agents.

Chapter 3

Functional Deployment Models and Call Flows

This chapter covers the following subjects:

- **Functional deployment models:** Standalone, Call Director, Comprehensive, VRU-only.
- **Detailed call flows:** Call flow details for each Functional Deployment Model.
- **Unified CCE interactions:** Network VRUs in Cisco Intelligent Contact Management Enterprise and IP originated calls with Cisco Unified Communications Manager.

Functional Deployment Models

As discussed briefly in the previous chapter, Unified CVP has some well-defined and commonly referenced functional deployment models. This chapter provides more details for these models and examines what each model accomplishes, its call flow, and even which native and non-native components are used. Understanding these models is critical to designing, implementing, and troubleshooting a Unified CVP solution. The chapter ends with a discussion that encompasses Unified CVP and its relationship and interactions with Unified CCE components, such as Unified ICM and Cisco Unified Communications Manager.

Standalone Model

Although this model is the simplest of all the functional deployment models, the flexibility available in Unified Call Studio applications is astounding. This model gives an organization the capability to replace legacy IVR systems using applications built using Unified Call Studio and hosted via the Unified CVP VoiceXML Server. The standalone model provides a standalone, automated self-service IVR solution that callers can access via TDM and VoIP terminating at Unified CVP's Ingress voice gateways. In addition callers could also access this solution via VoIP endpoints. Figure 3-1 shows the components used with this solution and their protocols.

Table 3-1 identifies which components are required, optional, and not used by this model. In addition, the Native column identifies components native to the Unified CVP solution.

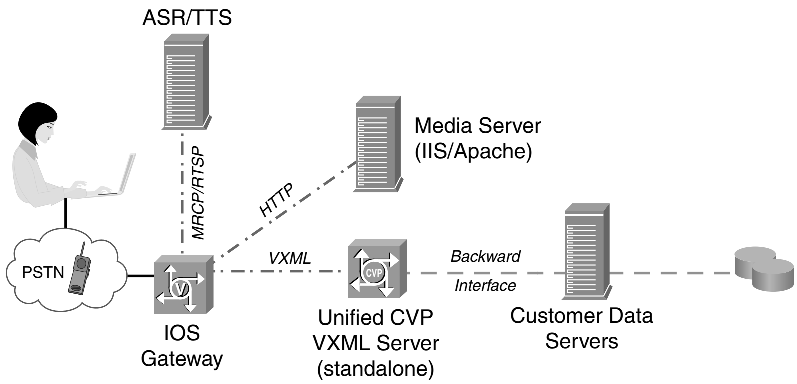


Figure 3-1 *Unified CVP Standalone Functional Deployment Model*

> **Note** Although Figure 3-1 does not show all the optional components, this model can also use content service switches to load balance both media fetch requests and VoiceXML connections requested from the VoiceXML Gateway. In addition, transfers such as bridged, blind, and even release trunk transfer types (TNT, hookflash, TBCT, and SIP REFER) are supported in this model, which also means that both Egress Gateways and CUCM endpoints are also optional, depending entirely on the customer call flow requirements.

Component and Protocol-Level Call Flow

This section examines how components interact with each other at the protocol level. Figure 3-2 illustrates the steps of a typical standalone call flow with each step detailed.

Following are the details of each step referenced in Figure 3-2:

Step 1. The call arrives from either the PSTN or a VoIP connection to the gateway. In this illustration, the gateway is functioning as both an Ingress Gateway and as a VoiceXML Gateway. The reason an Ingress Gateway is optional in Table 3-1 is because this initial call could be just a VoIP connection, which would require only a VoiceXML Gateway to terminate it and kick off the self-service application.

Step 2. The gateway sends an HTTP request to the Unified CVP VoiceXML Server. Prior to this occurring, the gateway performs the following actions:

a. The gateway matches an incoming Dialed Number Identification Service (DNIS) against its dial-peer configuration and kicks off a preconfigured application dial-peer. Example 3-1 provides a sample of how this dial-peer is configured for DNIS 1931 on the VoiceXML Gateway:

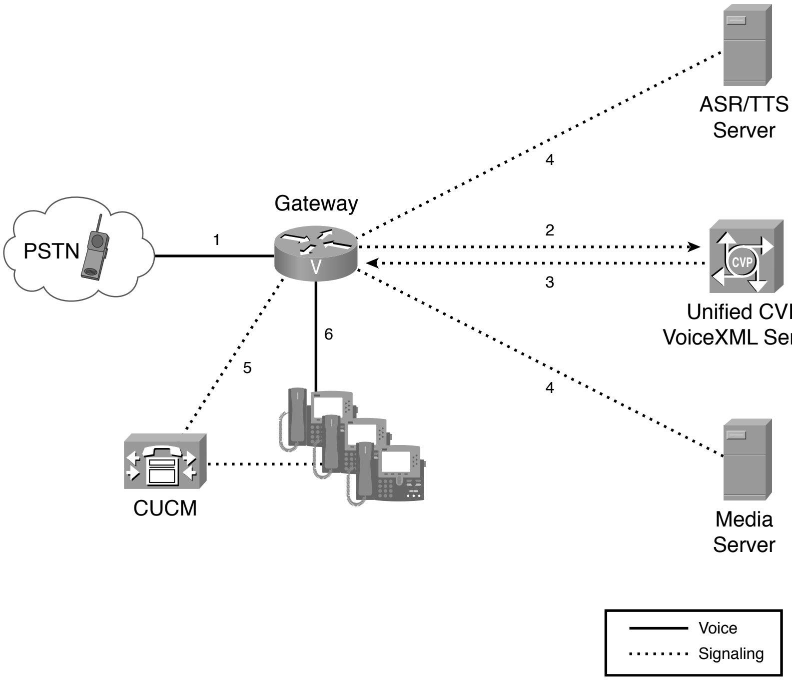


Figure 3-2 *Unified CVP Standalone Call Flow*

Example 3-1 *Incoming Dial-Peer for Kicking Off a Self-Service Application*

```
dial-peer voice 1 voip
 description —— Self Service SIP Calls from IP ——
 service myapp
 codec g711ulaw
 incoming called-number 1931
 dtmf-relay rtp-nte h245-signal h245-alphanumeric
 no vad
```

b. The application dial-peer invokes a self-service TCL script located in the router's flash memory, which invokes the Unified CVP standalone bootstrap VoiceVXML document. Example 3-2 provides a sample of how this is configured on the VoiceXML Gateway.

Example 3-2 *Self-Service Application Configuration on Gateway*

```
application
 service myapp flash:CVPSelfService.tcl
  paramspace english language en
  paramspace english index 0
  param CVPSelfService-port 7000
  param CVPSelfService-app MyApp
```

```
  param CVPPrimaryVXMLServer 192.168.1.100
  param CVPSecondaryVXMLServer 192.168.1.101
  paramspace english location flash
  paramspace english prefix en
 !
service CVPSelfService flash:CVPSelfServiceBootstrap.vxml
!
```

Table 3-1 *Unified CVP Standalone Native and Non-Native Component Usage*

Component	Required	Optional	Not Used	Native
SIP Service (Call Server)	—	—	Yes	Yes
IVR Service (Call Server)	—	—	Yes	Yes
ICM Service (Call)	—	Yes		Yes
H323 Service (Call Server)	—	—	Yes	Yes
VoiceXML Server	Yes	—	—	Yes
Unified Call Studio	Yes	—	—	Yes
Ingress Gateway		Yes	—	—
VXML Gateway	Yes	—	—	—
SIP Proxy	—	—	Yes	—
Gatekeeper	—	—	Yes	—
Operations Console	Yes	—	—	Yes
Reporting Server	—	Yes	—	Yes
ASR/TTS	—	Yes	—	—
Media Server	—	Yes	—	—
DNS Server	—	Yes	—	—
Content Services Switch	—	Yes		—
Unified ICM	—	Yes	—	—
Unified Call Manager	—	Yes	—	—
Egress Gateway	—	Yes	—	—

Note Example 3-2 also exposes the configuration for primary and secondary VoiceXML Servers using a parameter field. This approach illustrates an important consideration when deploying more than two standalone VoiceXML Servers. When more than two Unified CVP VoiceXML Servers are deployed, the VoiceXML Gateways must either be manually configured to use different pairs as primary and secondary servers, or a content load balancer switch would need to be deployed. If the latter approach is taken, the IP addresses configured in Example 3-2 would simply be the VIP addresses for the service load balanced by the content load balancers.

c. This VoiceXML document, also located in the router's flash memory, performs an HTTP request to the configured IP address of the Unified CVP VoiceXML Server. This IP address is preconfigured as a parameter when the self-service application is set up (refer to Example 3-2).

Step 3. The Unified CVP VoiceXML Server runs the application specified in the HTTP URL provided in Step 2. The result of running the application returns a dynamically generated VoiceXML document to the VoiceXML Gateway. Prior to building this dynamic VXML document, the Unified CVP VoiceXML Server can access backend systems as instructed by the application to incorporate personalized data into the VoiceXML document.

Step 4. The VoiceXML Gateway parses and renders the VoiceXML document. Following are the details pertaining to the rendering of this VoiceXML document:

a. If the VoiceXML instructions require a media file fetch operation, prior to fetching a prerecorded media file from a media server, the gateway first determines if the required prompt file is already cached in the gateway's http client cache. If so, that file is played to the caller; if not, the gateway resolves the media server name located in the fetch URL provided by the VoiceXML document. This media server name can either be a locally configured ip host entry or resolved via DNS.

b. If the VoiceXML instructions require the use of an ASR or TTS server, the VoiceXML Gateway sets up a connection to an ASR/TTS Server and streams media from the server. Caller input can be captured via DTMF detection on the Ingress Gateway or via DTMF/speech recognition on an ASR Server.

c. As defined by the VXML document, the VoiceXML submits an HTTP request containing the results of the caller input to the Unified CVP VoiceXML Server. The Unified CVP VoiceXML Server again runs the application specified in the HTTP request URL, passing it the results provided by the previous VoiceXML Gateway request and dynamically generates another VoiceXML document for rendering by the VoiceXML Gateway. This dialog continues until either the call is deemed as treated by the Unified Call Studio application or the caller terminates the call.

Step 5. The Ingress Gateway can, optionally, transfer the call to any destination that it can deliver a call to, such as Cisco Unified Communications Manager (CUCM).

Step 6. UCM can set up the call between an agent phone and the Ingress Gateway using a specific agent DNIS or a DNIS that corresponds to a hunt group or IP-IVR port. For this call flow this transfer is purely bridged, blind, or released trunk in nature and has no agent selection intelligence other than what CUCM can provide.

A slight variant of this call flow model is the use of Unified ICM to provide a lookup and return a label via the Unified CVP PG integration. Figure 3-3 illustrates this call flow followed by a detailed discussion about the involved steps.

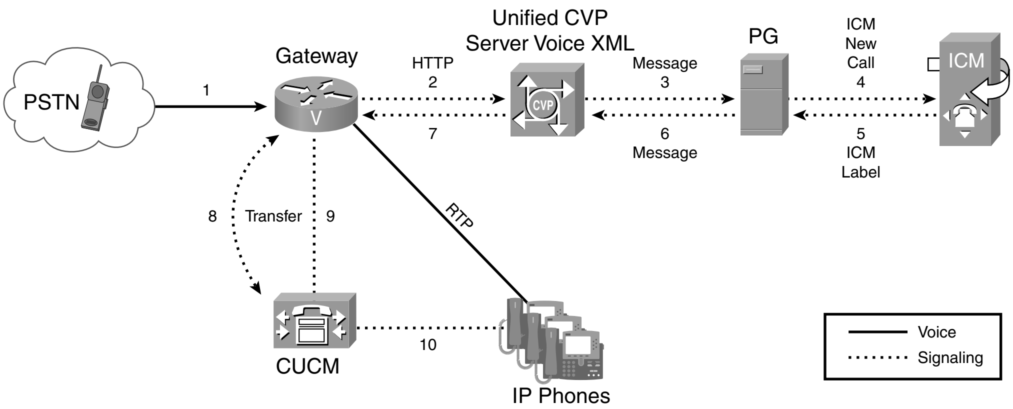


Figure 3-3 *Unified CVP Standalone with ICME Lookup Call Flow*

Step 1. The call arrives from either the PSTN or a VoIP connection to the gateway. In this illustration, the gateway is functioning as both an Ingress Gateway and as a VoiceXML Gateway.

Step 2. The gateway sends an HTTP URL request to the Unified CVP VoiceXML Server. The same configurations and substeps apply that were covered in the previous call flow.

Step 3. As a result of executing the application hosted by the Unified CVP VoiceXML Server, the server sends a message to the Unified CVP Call Server requesting that Unified CVP requests a label from Unified ICM.

Step 4. The Unified CVP Server sends Unified ICM a new call request via its Voice Response Unit (VRU) Peripheral Interface Manager (PIM), which is configured and hosted by the Peripheral Gateway (PG). This new route request invokes a new incoming route response that in turn invokes a routing script in Unified ICM.

Step 5. Unified ICM returns a Unified ICM routing label to the Unified CVP Call Server via the Unified CVP VRU PIM hosted by the PG.

Step 6. The Unified CVP Server returns a message to the VoiceXML Server with the routing label returned by Unified ICM.

Step 7. As in the previous call flow, the VoiceXML returned to the VoiceXML Gateway can include references to ASR/TTS, Media Servers for playing media, and the collections of digits or can be transfer instructions based on the Unified ICM label.

Step 8. The Ingress Gateway can, optionally, transfer the call to any destination that it can deliver a call to, such as CUCM.

Step 9. The Ingress Gateway signals the CUCM server for connection to either an IP Phone or IP IVR Port.

Step 10. CUCM can set up the call between an agent phone and the Ingress Gateway using a specific agent DNIS or a DNIS that corresponds to a hunt group or IP-IVR port. For this call flow this transfer is purely bridged, blind, or released trunk in nature and has no agent selection intelligence other than what CUCM can provide.

Caution This particular call flow returns only a routing label from the results of executing a routing script invoked by the new call dialog between Unified ICM and CVP. Running an external script or providing queuing is not supported with this variant. If call queuing is a requirement, you should use the comprehensive call flow model. In addition, this variant assumes the following:

- A Unified CVP Call Server has been defined using the Operations Console.
- A Unified Call Studio application has been created that contains a Unified ICM request label element.
- A Unified ICM script must be set up to handle the new call dialog request returning a correct routing label to Unified CVP.

The Unified Call Studio application must be deployed on the Unified CVP VoiceXML Server.

Call Flow Ladder Diagram

Figure 3-4 illustrates the call flow by showing the interaction between native and non-native CVP components. In addition, both call flows previously discussed are illustrated. However, the interaction with Unified ICM would not exist for the Unified CVP Standalone without ICME lookup call flow.

By examining the previous ladder diagram, it is obvious that the Unified CVP call server services such as the Session Initiation Protocol (SIP), H.323, or Interactive Voice Response (IVR) service are not used with this deployment model. The Intelligent Contact Management (ICM) service is engaged only in the execution of a Unified ICM lookup, and at no point in the call flow does CVP have any call control responsibilities.

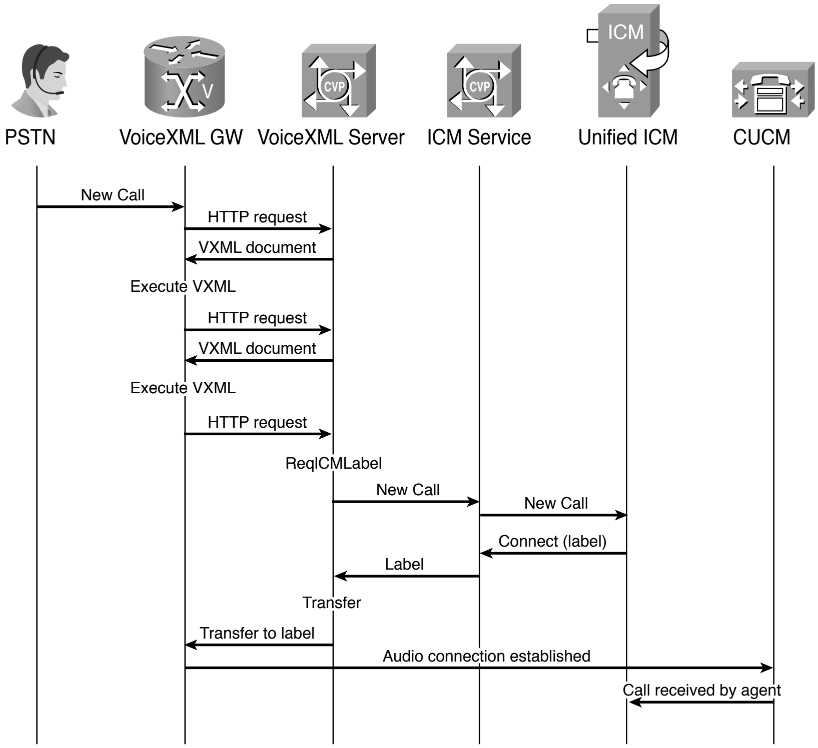


Figure 3-4 *Unified CVP Standalone Call Flow Ladder Diagram*

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, IVR application data cannot be passed to the new endpoints with this deployment model. Therefore, there will be no agent screen pop if the endpoint is a TDM ACD.

As noted earlier, this model supports the following types of transfers:

VoiceXML Bridged Transfer: The outcome of the transferred leg (that is, transfer failed, transfer call leg released, and so forth) is submitted back to the Unified CVP VoiceXML Server. The VoiceXML session is then resumed, and further iterations of the IVR call treatment and transfers can be performed. During the period of time that a call is transferred, a Unified CVP VXML Server port license is used if it is a bridged transfer.

VoiceXML Blind Transfer: With VoiceXML 2.0 Blind Transfers, the call remains connected through the Ingress Voice Gateway, but Unified CVP does not have any method to provide any subsequent call control.

Release Trunk Transfer (TNT, hookflash, TBCT, SIP Refer): As with VoiceXML 2.0 Blind Transfers, the Ingress Gateway port is released, and no subsequent call control is possible.

Note The VoiceXML transfers are invoked using Cisco Unified Call Studio's **transfer** element. Release Trunk Transfers are invoked by providing specifically formatted return values in Cisco Unified Call Studio's **subdialog_return** element.[1]

The Call Director Model

The purpose of this functional deployment model is to give organizations the ability to route and transfer calls across their existing VoIP networks. Because this functional deployment is often found in organizations preparing for or migrating to a VoIP contact center, it is no surprise that its strengths lie in its capability to switch calls to multiple TDM-based ACDs and IVRs without having to use PSTN prerouting or release trunk transfer services. When the organization is ready to implement CVP-based IVR services and Cisco Unified Contact Center Enterprise, it can migrate its Unified CVP deployment to the comprehensive functional deployment model discussed later in this chapter.

Furthermore, this particular deployment model gives Unified CVP and Unified ICM the capability to pass call data between these ACD and IVR locations. Unified ICM can also provide cradle-to-grave reporting for all calls. Although a customer can have a Unified CVP Reporting Server in this deployment model, it is optional because there is little call information stored in the Unified CVP reporting database. Both TDM and VoIP call originations are supported in this deployment model.[2] Figure 3-5 shows the components used with this solution and their protocols.

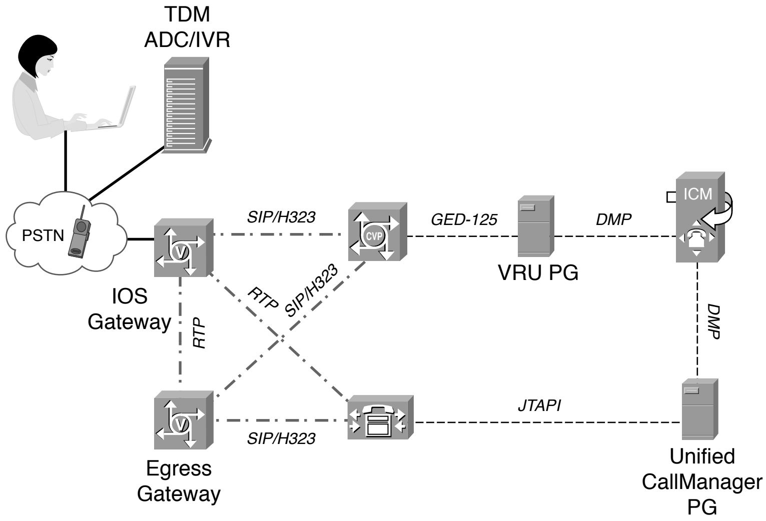


Figure 3-5 *Unified CVP Call Director Functional Deployment Model*

Table 3-2 identifies which components are required, optional, and not used by this model. In addition, the Native column identifies components that are native to the Unified CVP solution.

Table 3-2 *Unified CVP Call Director Native and Non-Native Component Usage*

Component	Required	Optional	Not Used	Native
SIP Service (Call Server)	Yes (if SIP)	—	—	Yes
IVR Service (Call Server)		—	Yes	Yes
ICM Service (Call Server)	Yes	—	—	Yes
H323 Service (Call Server)	Yes (If H323)	—	—	Yes
VoiceXML Server	—	—	Yes	Yes
Unified Call Studio	—	—	Yes	Yes
Ingress Gateway	Yes	—	—	—
VXML Gateway		—	Yes	—
SIP Proxy	Yes (if SIP)	—	—	—
Gatekeeper	Yes (If H323)	—	—	—
Operations Console	Yes	—	—	Yes
Reporting Server	—	Yes	—	Yes
ASR/TTS	—	—	Yes	—
Media Server	—	—	Yes	—
DNS Server	—	Yes	—	—
Content Services Switch	—	—	Yes	—
Unified ICM	Yes	—	—	—
Unified Call Manager	—	Yes	—	—
Egress Gateway	—	Yes	—	—

Note Although Figure 3-5 does not show all the optional components, this model can use SIP proxy servers and gatekeepers depending on what call control protocol is used in the deployment.

SIP-Based Protocol-Level and Component Call Flow

This section examines how the components interact with each other at the protocol level. As illustrated for the Unified CVP Standalone model, Figure 3-6 reviews the steps of a typical call director call flow.

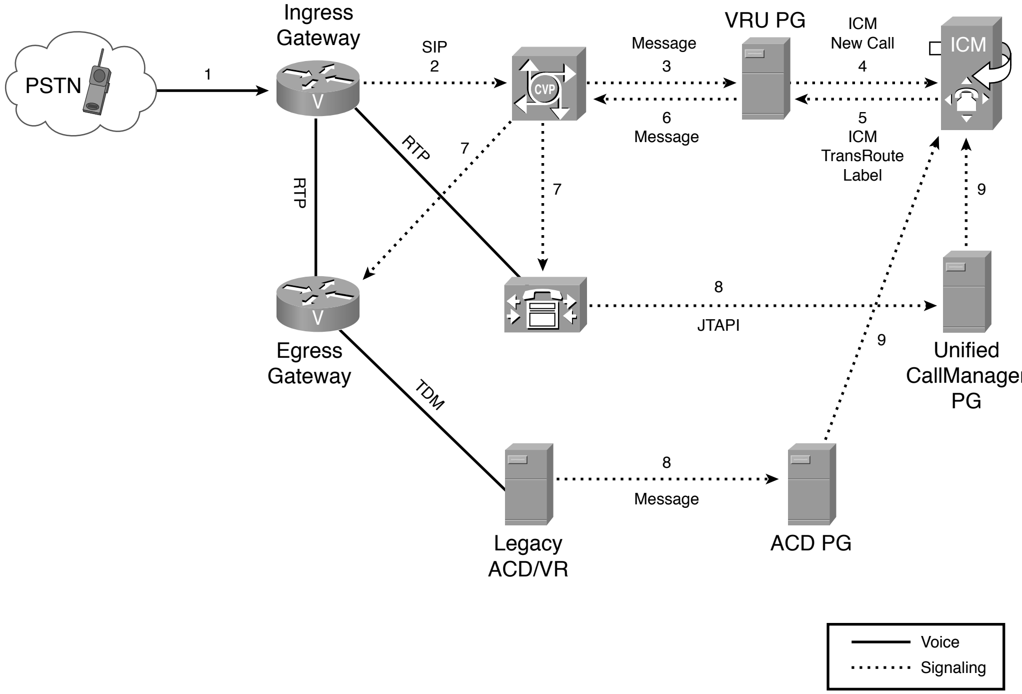


Figure 3-6 *Unified CVP Call Director SIP Call Flow*

Step 1. The call arrives from either the PSTN or a VoIP connection to the gateway.

Step 2. The Ingress Gateway sends a SIP INVITE message either directly to the Unified CVP call server or to a SIP Proxy Server, which forwards the request to the Unified CVP Server SIP Service.

Step 3. The Unified CVP Server SIP Service sends a route request to Unified ICM via the Unified CVP Server ICM Service and the PG.

Step 4. The Unified CVP Server sends Unified ICM a new call request via its VRU PIM that is configured and hosted by the PG. This new call request invokes a new incoming dialed number that in turn invokes a routing script in Unified ICM.

Step 5. The Unified ICM routing script selects a target and returns a translation route label to the PG via the Unified CVP VRU PIM hosted by the PG.

Step 6. The Unified CVP Server's ICM Service processes the instructions provided by the VRU PIM and hands the label over to the SIP Service for call setup.

Step 7. The Unified CVP Call Server's SIP Service signals either the Ingress Gateway or the proxy server, depending on its configuration. This enables the call to be set up between either an Egress Gateway or a Unified Communications Manager cluster. Depending on the solution requirements, the call server can connect calls to either an Egress Gateway or Unified Communications Manager. As depicted in Figure 3-6, Real-time Transport Protocol (RTP) or voice bearer traffic flows directly between the Ingress Gateway and an Egress Gateway or a Unified Communications Manager IP Phone. Call control signaling continues to flow through the Unified CVP to enable subsequent call control.

Step 8. When the call arrives at the selected termination, the termination equipment sends a request to its PG for routing instructions.

Step 9. When the call arrives at the selected termination, the termination equipment sends a request to its PG for routing instructions. This step involves the translation route and enables any call data from the previously run Unified ICM script to be passed to the selected termination. In the case of a TDM-based IVR, the self-service can occur with the caller either being released or transferred to a live agent. In the case of a TDM-based ACD, the call may be queued until an agent is available.

Note In either of these cases, the IVR self-service or ACD treatment is not handled by Unified CVP but configured and delivered via the legacy TDM-based IVR or ACD. VRU Scripts and transfer to a VRU leg are not available in this call flow model.

VoIP Transfers Using SIP

Figure 3-7 illustrates how VoIP transfers using SIP are accomplished within this model. Because the Unified CVP call server still maintains call control, it has the capability to signal the Ingress Gateway to move the call from one termination point to another. This is accomplished via PG messages from the original termination point, which may have been a legacy ACD/IVR or Unified Communications Manager. Although Figure 3-7 illustrates this transfer using a second Egress Gateway, this transfer could occur to the same or a different Egress Gateway or Unified Communications Manager cluster.

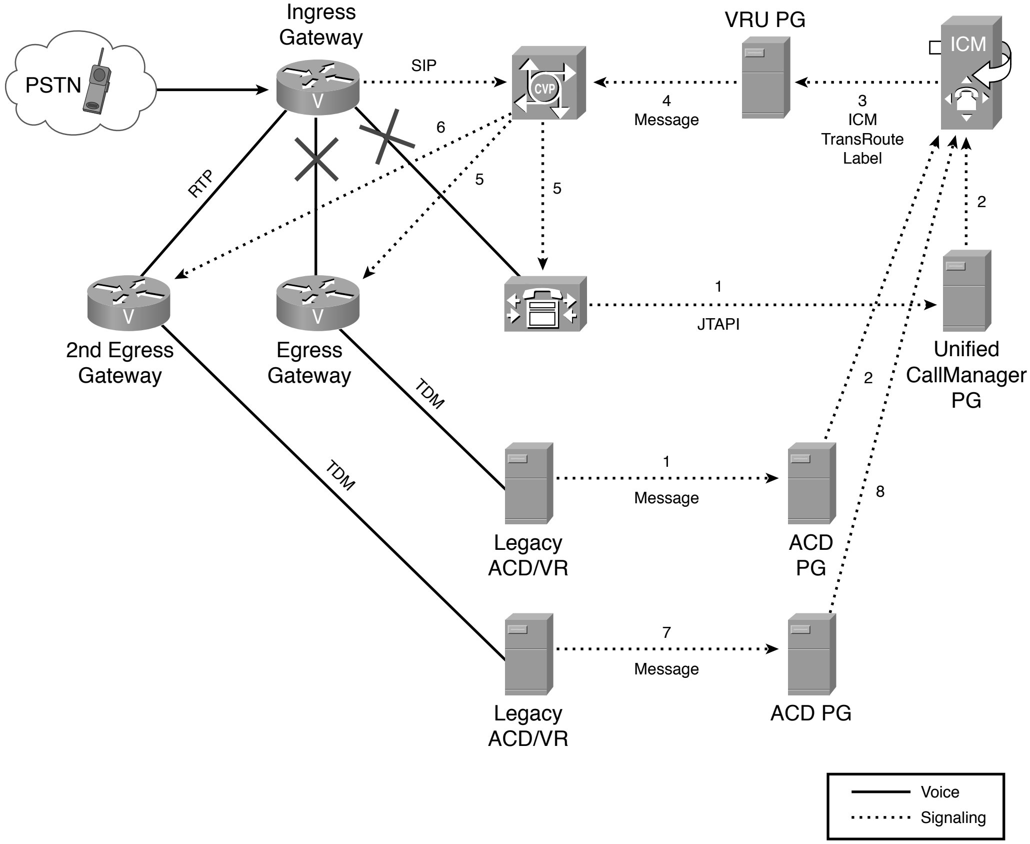


Figure 3-7 *Unified CVP Call Director SIP Transfers Call Flow*

Following are the details for the steps previously referenced:

Step 1. A caller from a previously routed call, which is currently controlled by the Unified CVP Call Server, requests to be transferred to another location.

Step 2. The TDM IVR, ACD, or Unified Communications Manager sends a post-route request with call data (via its PG) to Unified ICM.

Note Regardless of whether the call was initially routed to a TDM IVR, ACD, or Unified Communications Manager cluster location, the call can request to be transferred to another location as long as the Unified CVP Call Server still has call control with the Ingress Gateway.

Step 3. When Unified ICM receives this post-route request, it runs an associated routing script based on the transferred dialed number. The Unified ICM routing script selects a target and returns a translation route label to the PG via the Unified CVP VRU PIM hosted by the PG.

Step 4. The Unified CVP Server's ICM Service processes the instructions provided by the VRU PIM and hands the label over to the SIP Service for call setup.

Step 5. The Unified CVP Server's SIP Service releases the call leg to the originally selected termination devices. In Figure 3-7, these devices were either an Egress Gateway or a Unified Communications Manager.

Step 6. The Unified CVP Call Server's SIP Service signals either the Ingress Gateway or the proxy server, depending on its configuration, which enables for the call to be set up between either the second Egress Gateway or a different Unified Communications Manager cluster. The call can also be extended to the same devices that the originating call terminated on. However, Steps 1 through 6 are still required. Existing RTP streams are torn down and brought back up with the second termination device, as depicted in Figure 3-7.

Step 7. When the call arrives at the second termination device, the termination equipment sends a request to its PG for routing instructions.

Step 8. This step involves the translation route and enables any call data from the previously run Unified ICM script to be passed to the selected termination. In the case of a TDM-based IVR, self-service can occur with the caller either being released or transferred to a live agent. In the case of a TDM-based ACD, the call may be queued until an agent is available.

Note Calls can continue to be transferred between locations using the same VoIP-based transfer call flow previously described.

SIP Call Flow Ladder Diagram

Figure 3-8 illustrates the call flow showing the interaction between native and non-native CVP components. Expanding from Figure 3-3, additional services are engaged with this deployment model than what was illustrated for the standalone model. For example, the SIP Service is now used with a SIP proxy server, Egress Gateways, and even a Unified Communications Manager cluster.

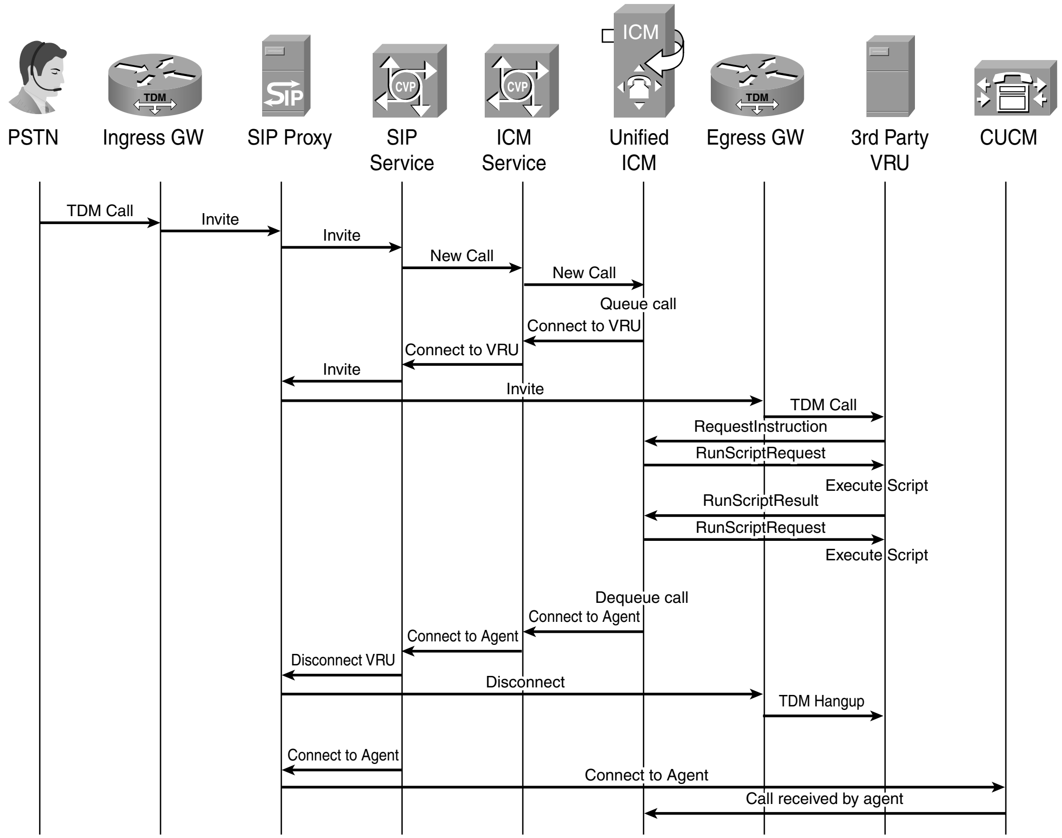


Figure 3-8 *Unified CVP Call Director SIP Call Flow Ladder Diagram*

> **Note** Figure 3-8 also depicts the overlapping agent connect conversations, which are mutually exclusive and presented on the same ladder diagram to provide a comparison between a transfer that occurs to an Egress Gateway versus the same transfer to a Unified Communications Manager cluster with agent IP Phones.

H.323 Protocol-Level and Component Call Flow

Although it has been mentioned a few times in this book that H.323 is still supported for upgrades and not for green-field deployments, you need to understand the basic protocol call flow and component interaction for H.323. Migrations require engineers to migrate these call flows to SIP. Understanding how they currently operate can provide important insight for performing migrations. Figure 3-9 illustrates the H.323 call flow, which has striking similarities to the previous SIP call flow shown earlier in Figure 3-6.

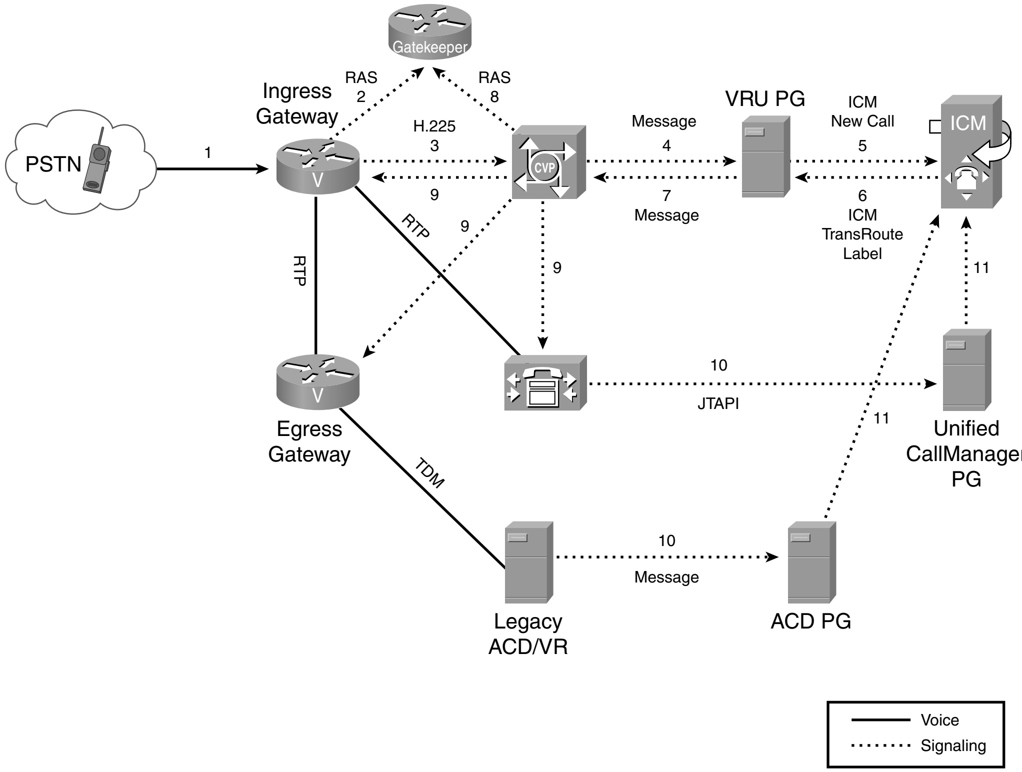


Figure 3-9 *Unified CVP Call Director H.323 Call Flow*

Following are the details for the steps previously referenced:

Step 1. The call arrives from either the PSTN or a VoIP connection to the gateway.

Step 2. The Ingress Gateway sends an H.225 Registration, an Admission, and a Status (RAS) request to the H.323 gatekeeper to find the IP address of an appropriate Unified CVP Server for the incoming dialed number.

Step 3. The Ingress Gateway sends an H.225 call setup message to the Unified CVP Server's H.323 Service.

Note During the initial call setup between the Ingress Gateway and the CVP Call Server, a brief G.711 voice stream exists and is immediately torn down when the Unified CVP Call Server gains call control.

Step 4. The Unified CVP Server's H.323 Service sends a route request to Unified ICM via the Unified CVP Server's IVR Service, the Unified CVP Server's ICM Service, and the PG.

Step 5. The Unified CVP Server sends Unified ICM a new call request via its VRU PIM that is configured and hosted by the PG. This new call request invokes a new incoming dialed number that in turn invokes a routing script in Unified ICM.

Step 6. The Unified ICM routing script selects a target and returns a translation route label to the PG via the Unified CVP VRU PIM hosted by the PG.

Step 7. The Unified CVP Server's ICM Service processes the instructions provided by the VRU PIM and hands the label over to the H.323 Service for call setup.

Step 8. The Unified CVP Call Server's H.323 Service sends a RAS request to the H.323 gatekeeper to find the IP address of the appropriate termination (an Egress Voice Gateway to the PSTN, an Egress Voice Gateway front-ending a TDM peripheral or a Unified Communications Manager Cluster).

Step 9. The Unified CVP Server's H.323 Service then sends an H.225 call setup message to the termination location (Egress Voice Gateway or Unified Communications Manager cluster) and makes an Empty Capability Set (ECS) request to the Ingress Voice Gateway to redirect the call. RTP or voice bearer traffic flows directly between the Ingress Gateway and the selected termination point (refer to Figure 3.9). Call control signaling continues to flow through the Unified CVP call server to allow subsequent call control.

Step 10. When the call arrives at the selected termination, the termination equipment sends a request to its PG for routing instructions.

Step 11. This step involves the translation route and enables any call data from the previously run Unified ICM script to be passed to the selected termination. In the case of a TDM-based IVR, self-service can occur with the caller either being released or transferred to a live agent. In the case of a TDM-based ACD, the call may be queued until an agent is available.

Note Although a SIP Proxy Server is optional for both the switch and VRU leg of a SIP call with Unified CVP, an H.323 gatekeeper is not for H.323 calls. The H.323 service running on a Unified CVP Call Server requires a registration with a gatekeeper; without this configuration and subsequent registration, the call server from an H.323 perspective remains in a down state. The implication of this requirement on SIP and H.323 call flows is that a SIP configuration can exclude the use of a SIP proxy server relying on static SIP routes, whereas an H.323 configuration requires a gatekeeper to have the call server up and processing H.323 calls. Even with edge queuing architectures that use the H.323 Set Transfer Label configuration, an active gatekeeper registration must still exist on the call server for the label to be sent back to the originating H.323 Gateway, even though the architecture ignores the gatekeeper lookup during the VRU leg of a H.323 call.

VoIP Transfers Using H323

Figure 3-10 illustrates how VoIP transfers work with H.323. The call flow is similar to SIP; however, a gatekeeper lookup using RAS is required in the middle of the call flow.

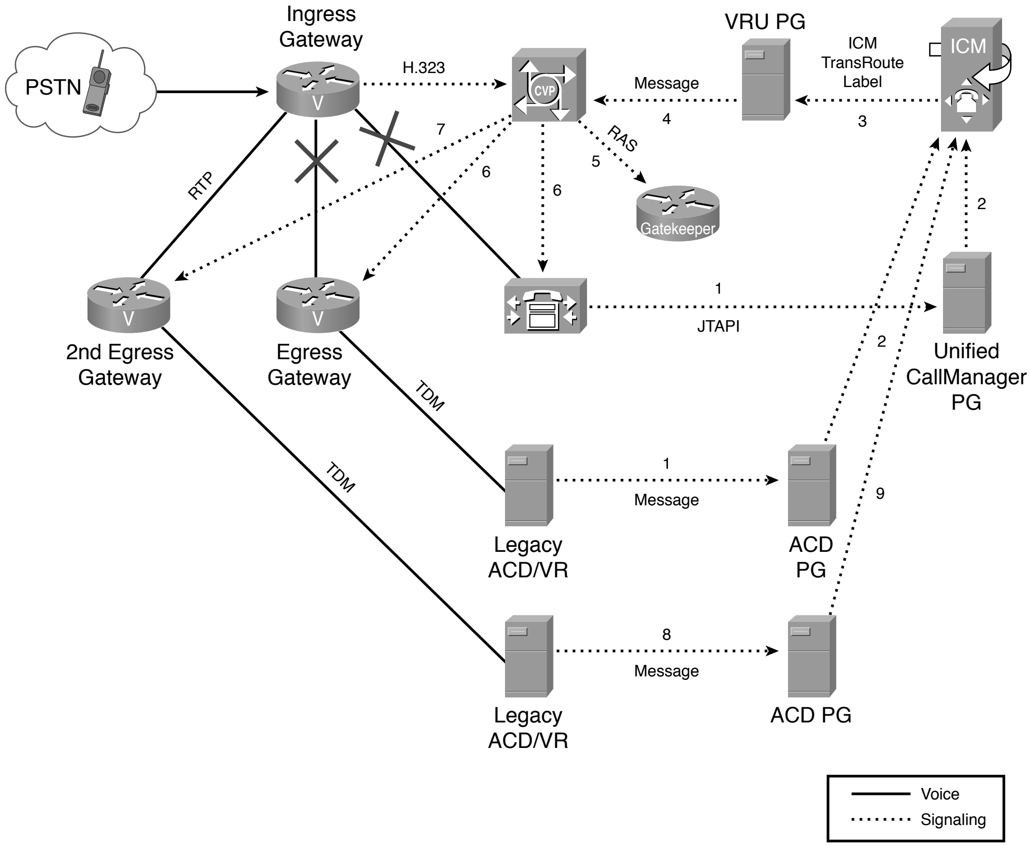


Figure 3-10 *Unified CVP Call Director H.323 Transfers Call Flow*

Following are the details for the steps previously referenced:

Step 1. A caller from a previously routed call, which is currently controlled by the Unified CVP Call Server, requests to be transferred to another location.

Step 2. The TDM IVR, ACD, or Unified Communications Manager sends a post-route request with call data (via its PG) to Unified ICM.

Note Regardless of whether the call was initially routed to a TDM IVR, ACD, or Unified Communications Manager cluster location, the call can request to be transferred to another location as long as the Unified CVP Call Server still has call control with the Ingress Gateway.

Step 3. When Unified ICM receives this post-route request, it runs an associated routing script based on the transferred dialed number and other criteria. The

Unified ICM routing script selects a target and returns a translation route label to the PG via the Unified CVP VRU PIM hosted by the PG.

Step 4. The Unified CVP Server's ICM Service processes the instructions provided by the VRU PIM and hands the label over to the H.323 Service for call setup.

Step 5. The Unified CVP Server's H.323 Service queries the H.323 gatekeeper to get an IP address for the new termination.

Step 6. The Unified CVP Server's H.323 Service releases the call leg to the originally selected termination devices. These devices were either an Egress Gateway or a Unified Communications Manager (refer to Figure 3-10).

Step 7. The Unified CVP Call Server's H.323 Service signals the original termination device, which enables the call to be set up between either the second Egress Gateway or a different Unified Communications Manager cluster. The call could also be extended to the same devices that the originating call terminated on. However, Steps 1 through 6 are still required. Existing RTP streams are torn down and brought back up with the second termination device (refer to Figure 3-10).

Step 8. When the call arrives at the second termination device, the termination equipment sends a request to its PG for routing instructions.

Step 9. This step involves the translation route and enables any call data from the previously run Unified ICM script to be passed to the selected termination. In the case of a TDM-based IVR, self-service can occur with the caller either being released or transferred to a live agent. In the case of a TDM-based ACD, the call may be queued until an agent is available.

Note Calls may continue to be transferred between locations using the same VoIP-based transfer call flow previously described.

Transfers and Subsequent Call Control

In addition to the transfers managed by Unified ICM, the Call Director Deployment model can transfer calls to non-ICM terminations or invoke a Release Trunk Transfer to the PSTN. However, if a call is transferred to a non-ICM controlled termination, call data cannot be passed to the termination, further call control is impossible for the call, and the cradle-to-grave call reporting that Unified ICM gathers is complete. In the case of a Release to Trunk Transfer on the Ingress Voice Gateway, call data or call control cannot be maintained. However, if the call is a translation routed to another ICM peripheral, call data and cradle-to-grave reporting can be maintained.

If a transfer fails or the termination device returns a 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 Re-query operation within Unified ICM, enabling a different target to be selected or execution of a remedial action.[3]

Comprehensive Model

This next function deployment model provides organizations with a mechanism to route and transfer calls across a VoIP network, offers IVR services, and queues calls before routing to a selected agent. These features are usually found in situations in which an organization is interested in providing a pure IP-based contact center. A caller can initially be directed into an IVR service, exit, and be queued for the next available agent. In addition, transfers are supported between Unified CCE Agents. The passing of data between Unified CVP and ICM is fully supported, and cradle-to-grave reporting for all calls is also supported. Figure 3-11 shows the components used with this solution and their protocols.

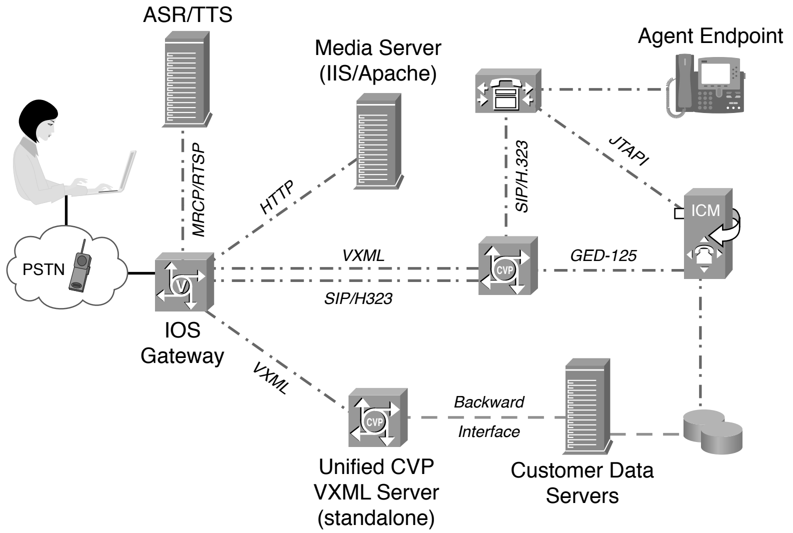


Figure 3-11 *Unified CVP Comprehensive Functional Deployment Model*

> **Note** Although some documentation identifies Basic Video as a potential fifth deployment model, Basic Video is nothing more than a comprehensive deployment model with video endpoints replacing voice endpoints; therefore, the call flows provided in this section also apply to the implementation of Basic Video with Unified CVP. The use of Cisco Unified Videoconferencing hardware, Radvision IVP, and Radvision iContact are not required for the implementation of the Basic Video Service and are out of the scope of this book.

Table 3-3 identifies which components are required, optional, and not used by this model. In addition, the Native column identifies components native to the Unified CVP solution.

Table 3-3 *Unified CVP Comprehensive Native and Non-Native Component Usage*

Component	Required	Optional	Not Used	Native
SIP Service (Call Server)	Yes	—	—	Yes
IVR Service (Call Server)	Yes	—	—	Yes
ICM Service (Call Server)	Yes	—	—	Yes
H323 Service (Call Server)	Yes (If H323)	—	—	Yes
VoiceXML Server	—	Yes	—	Yes
Unified Call Studio	—	Yes	—	Yes
Ingress Gateway	Yes	—	—	—
VXML Gateway	Yes	—	—	—
SIP Proxy	—	Yes	—	—
Gatekeeper	Yes (If H323)	—	—	—
Operations Console	Yes	—	—	Yes
Reporting Server	—	Yes	—	Yes
ASR/TTS	—	Yes	—	—
Media Server		Yes	—	—
DNS Server	—	Yes	—	—
Content Services Switch	—	Yes	—	—
Unified ICM	Yes	—	—	—
Unified Communications Manager	—	Yes	—	—
Egress Gateway	—	Yes	—	—

Note Although Table 3-3 lists the Unified Communications Manager and Egress Gateway as optional components, at least one of these components should exist to enable a call to be transferred to either a legacy ACD Agent via an Egress Gateway or to a Unified CCE Agent hosted by a Unified Communications Manager cluster. Which component is deployed depends entirely on the organization's requirements. In addition, the use of load balancers, media servers, and even an ASR/TTS Server are also optional because the VXML Gateways can be configured in such a way to allow them to operate without these components. In the case of ASR/TTS, this would be true only if the organization has no requirement to perform the ASR/TTS treatment for incoming calls.

SIP-Based Protocol-Level and Component Call Flow

This section examines the detailed call flow steps performed in a typical comprehensive call flow. As its name indicates this model is the most complex from a call flow perspective. Figure 3-12 illustrates a typical comprehensive call flow with a SIP proxy server as part of the solution. Figure 3-13, provided later in this section, covers a similar SIP call flow without a proxy server.

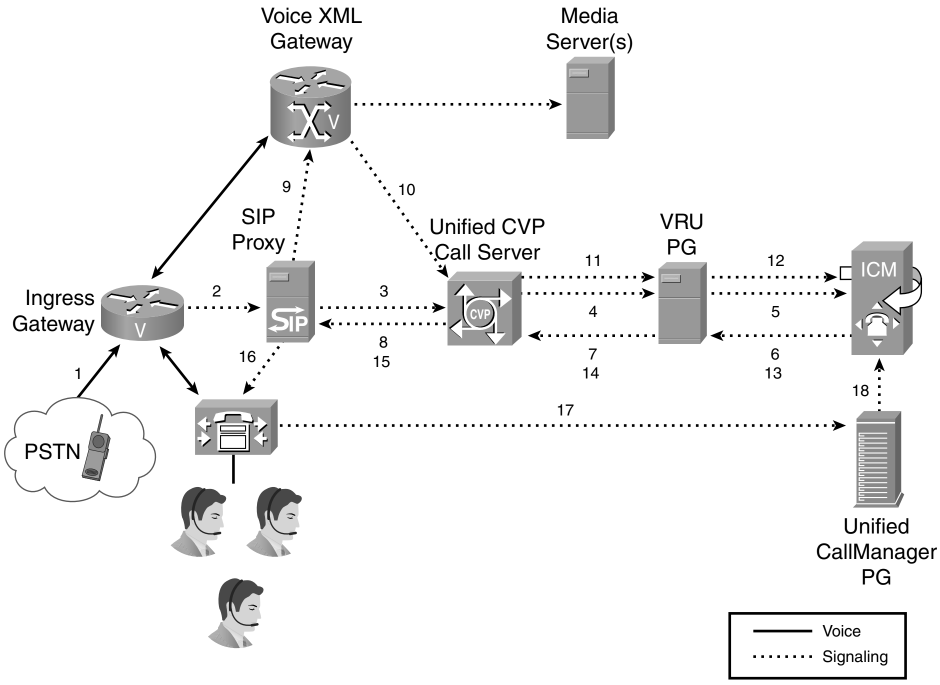


Figure 3-12 *Unified CVP Comprehensive SIP Call Flow with a SIP Proxy Server*

Following are the details for the steps previously referenced:

Step 1. The call arrives from either the PSTN or a VoIP connection to the gateway.

Step 2. The Ingress Gateway sends a SIP INVITE message the SIP Proxy Server.

Step 3. The SIP Proxy Server forwards this SIP INVITE to the Unified CVP Server's SIP Service.

Step 4. The SIP Service sends a new call request to Unified ICM via the Unified CVP Server ICM Service and the PG.

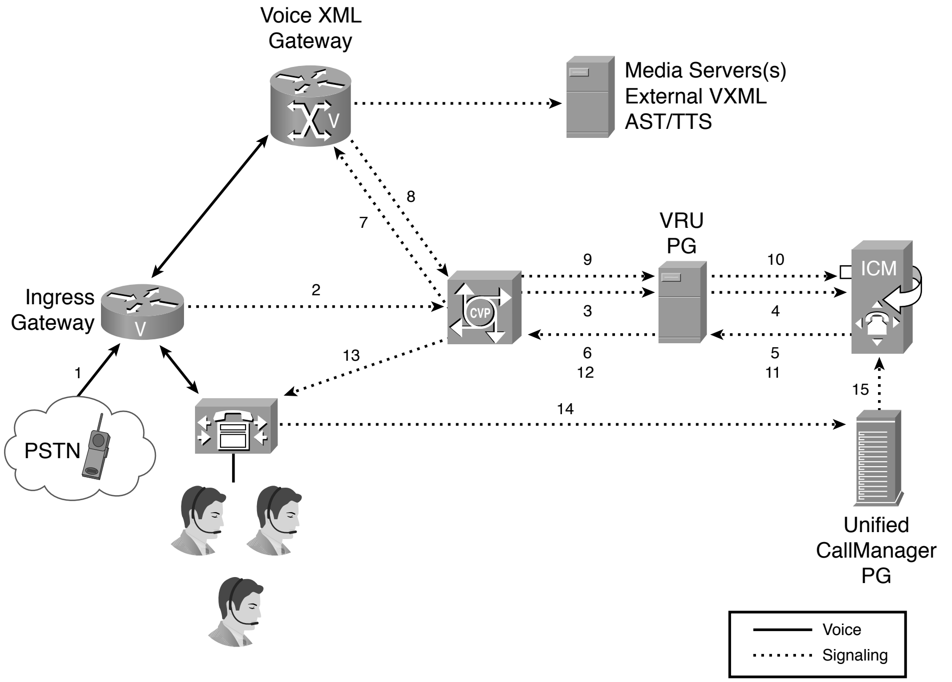


Figure 3-13 *Unified CVP Comprehensive SIP Call Flow Without a SIP Proxy Server*

Step 5. The Unified CVP Server sends Unified ICM a new call request via its VRU PIM that is configured and hosted by the PG. This new call request invokes a routing script in Unified ICM based on the dialed number provided by Unified CVP.

Step 6. The Unified ICM routing script determines that the caller must be transferred to the VRU and passes a Connect to VRU request to the PG, which will be forwarded to the ICM Service on the Unified CVP Call Server.

Step 7. PG passes the information provided in Step 6 to the ICM Service on the Unified CVP Call Server.

Step 8. Invitation to Connect to VRU goes from SIP Service back to the SIP Proxy Server requiring the SIP Proxy Server to determine which VXML Gateway based on its routing table should handle the Connect to VRU and subsequent call treatment session for the VRU leg of the call.

Step 9. Invitation (including information about Ingress Gateway) is sent from the SIP Proxy Server to a VXML Gateway, which then connects the audio path back to the Ingress Gateway.

Step 10. The VRU label causes the VXML Gateway to fire off an application dial-peer, which starts the VRU or application leg of the call. The VXML Gateway connects to the Unified CVP Call Server via HTTP and requests instructions for treating the connected call. This HTTP new call request is handled by the Unified CVP Server's IVR Service, which then passes this request to the Unified CVP Server's ICM Service.

Step 11. ICM Service sends Request Instructions to Unified ICM via the PG.

Step 12. Unified ICM continues the script that was started in Step 5 and processes additional nodes that produce Run Script requests to the Unified CVP Server's ICM Service. It hands these instructions over to the Unified CVP Server's IVR Service that converts them into VXML pages that are forwarded back to the VXML Gateway for rendering and execution. This process can also include prompt and collect instructions and continues between Unified ICM, Unified CVP, and the VXML Gateway until an agent becomes available. The VoiceXML Gateway also fetches any media files requested in the VXML pages from the media server (refer to the top of Figure 3-12).

Step 13. The agent becomes available, so Unified ICM dequeues the call and asks to be disconnected from the VXML Gateway. Unified ICM passes a connect-to-agent request to the Unified CVP Server's ICM Service via the VRU PG.

Step 14. PG passes the information provided in Step 13 to the ICM Service on the Unified CVP Call Server. The ICM Service passes this connect request to the Unified CVP Server's SIP Service.

Step 15. The Unified CVP Server's SIP Service passes this VRU disconnect request to the SIP Proxy Server.

Step 16. The SIP Proxy Server passes a disconnect message to the VXML Gateway. SIP Service passes a connect-to-agent request to the SIP Proxy Server. The SIP Proxy Server passes this connect-to-agent request back via a SIP INVITE to the Cisco Unified Communications Manager Subscriber server responsible for handling the configured SIP Trunk. Because the audio path was torn down between the Ingress Gateway and the VXML Gateway, a new one is established between the Ingress Gateway and the Unified CCE Agent IP Phone hosted on the CUCM cluster.

Step 17. Unified Communications Manager informs the PG that a call was delivered to a Unified CCE Agent.

Step 18. PG notifies Unified ICM that a call has been delivered to the Unified CCE Agent.

Note It is common for the Unified CCE Agent IP Phone to be registered with a subscriber server in the Cisco Unified Call Manager cluster that is not the same server configured to communicate with PG and ICM. The Computer Telephony Integration (CTI) Manager service, which should be running on all servers in the cluster that control agent IP Phones, is cluster-aware and forwards updates for agent state, and so on to the CTI Manager service running on the subscriber server connected to the Communications Manager PG via JTAPI. This enables a pair of Communications Manager PGs to service an entire CUCM cluster, even when IP Phones are registered to different servers in the cluster. However, there are still limitations on how many Unified CCE Agents can be serviced via a single pair of Communication Manager PGs. For additional agent and PG sizing considerations please refer to the Unified Contact Center Enterprise Solution Reference Network Design (SRND) or Bill of Materials located at Cisco.com.

Although a previous call flow detailed how the Comprehensive model uses a SIP Proxy Server, a SIP Proxy Server is an optional non-native component. This indicates that a second call flow can occur when a SIP Proxy is not part of the solution. Figure 3-13 provides a look at this call flow, which is similar to the previous call flow (refer to Figure 3-12).

Following are the details for the steps previously referenced:

Step 1. The call arrives from either the PSTN or a VoIP connection to the gateway.

Step 2. The Ingress Gateway sends a SIP INVITE message to the Unified CVP Call Server's SIP Service.

Step 3. The SIP Service sends a new call request to Unified ICM via the Unified CVP Server ICM Service and the VRU PG.

Step 4. The Unified CVP Server sends Unified ICM a new call request via its VRU PIM configured and hosted by the PG. This new call request invokes a new incoming dialed number that invokes a routing script in Unified ICM.

Step 5. The Unified ICM routing script determines that the caller must be transferred to the VRU and passes a Connect to VRU request to the PG, which will be forwarded to the ICM Service on the Unified CVP Call Server.

Step 6. PG passes the information provided in Step 5 to the ICM Service on the Unified CVP Call Server.

Step 7. The Unified CVP Server's SIP Service determines which VXML Gateway should handle the Connect to VRU and the subsequent call treatment session by examining it's local static SIP routing table configured on the all Server.

Step 8. The VRU label causes the VXML Gateway to fire off an application dial-peer, which starts the VRU or application leg of the call. The VXML Gateway connects to the Unified CVP Call Server via HTTP and requests instructions for treating the connected call. This HTTP new call request is handled by the Unified CVP Server's IVR Service, which then passes this request to the Unified CVP Server's ICM Service.

Step 9. The ICM Service sends Request Instructions to Unified ICM via the PG.

Step 10. Unified ICM continues the script that was started in Step 5 and processes additional nodes that produce Run Script requests to the Unified CVP Server's ICM Service, which then hands these instructions over to the Unified CVP

Server's IVR Service that converts them into VXML pages forwarded back to the VXML Gateway for rendering and execution. This process can also include prompt and collect instructions and continues between Unified ICM, Unified CVP, and the VXML Gateway until an agent becomes available. The VoiceXML Gateway also fetches any media files requested in the VXML pages from the media server (refer to the top of Figure 3-13).

Step 11. The agent becomes available, so Unified ICM dequeues the call and requests to be disconnected from the VXML Gateway. Unified ICM passes a connect-to-agent request to the Unified CVP Server's ICM Service via the PG.

Step 12. PG passes the information provided in Step 11 to the ICM Service on the Unified CVP Call Server. The ICM Service passes this connect request to the Unified CVP Server's SIP Service.

Step 13. The Unified CVP Server's SIP Service passes disconnect to the VXML Gateway. The SIP Service then passes the connect-to-agent request via a SIP INVITE to the Cisco Unified Communications Manager Server responsible for handling the configured SIP Trunk for the cluster. Because the audio path was torn down between the Ingress Gateway and the VXML Gateway, a new one is established between the Ingress Gateway and the Unified CCE Agent IP Phone hosted on the CUCM cluster.

Step 14. Unified Communications Manager informs the PG that a call was delivered to a Unified CCE agent.

Step 15. PG notifies Unified ICM that a call has been delivered to the Unified CCE Agent.

Caution The key takeaway from the previous call flow is that by choosing not to use a SIP Proxy Server, critical elements required for redundancy do not exist. By using a supported SIP Proxy Server, a solution can achieve a high-level of redundancy and increase efficiency pertaining to configuration management and troubleshooting. The call flow (refer to Figure 3-13) requires that each Unified CVP Call Server to be configured with static SIP routes for all VRU and transfer labels. In addition, the Ingress Gateways also must be configured with multiple dial-peers to enable it to choose which Unified CVP Call Server to connect via the switch leg of the call. Although you can use DNS to ease some of this configuration pain, it is best practice in medium or large Unified CVP deployments to use a supported SIP Proxy Server.

VoIP Transfers using SIP

Figure 3-14 illustrates how a VoIP transfer is handled when a Unified CCE agent initiates the transfer via its agent desktop application.

The following detailed steps cover two options supported in the configuration of this transfer: the use of an ICM Dialed Number Plan and Unified Communications Manager

Route Points. Both options follow the same call flow but are quite different in how they engage during a transfer.

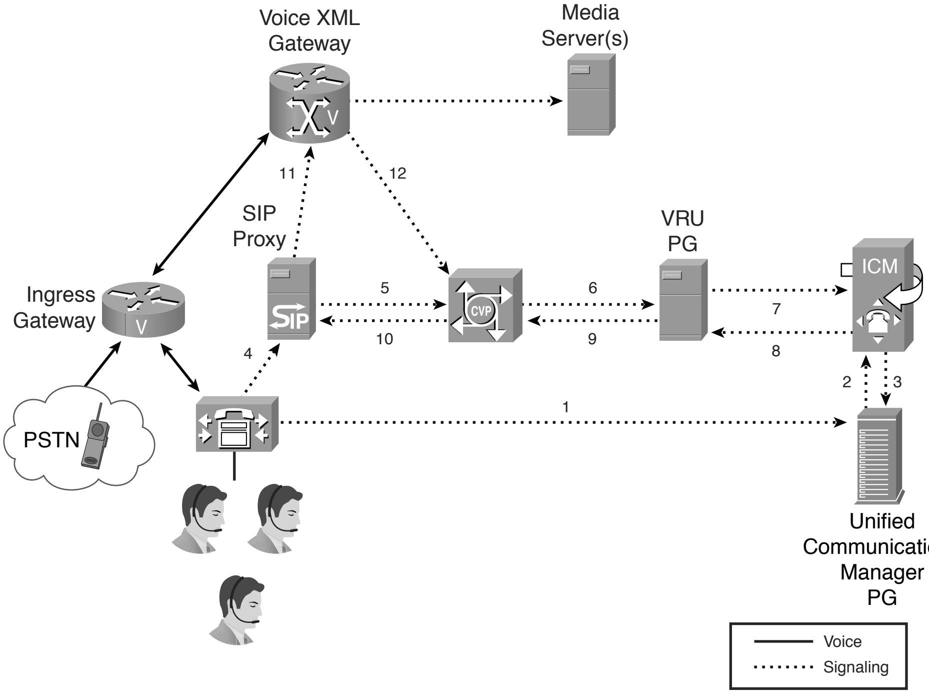


Figure 3-14 *Unified CVP Comprehensive SIP Transfer Call Flow Using an ICM Dialed Number Plan*

The ICM Dialed Number plan is configured within Unified ICM and enables a dial plan matrix to exist where translations can be configured and scripts can be executed before the Unified Communications Manager is asked to decide how to route the label returned by Unified ICM. This enables a transfer to be intercepted as soon as it is initiated on the agent desktop and keeps the dial plan for the contact center stored inside the Unified ICM database and not entirely within the CUCM database.

A second approach is to use Route Points configured within CUCM. This approach is far less desirable simply because of the amount of provisioning and that the dial plan exists both in Unified ICM and Unified Communications Manager. It increases the difficulty around provisioning and troubleshooting the solution.

Note The ICM Dialed Number Plan (DNP) can be used only if the Unified CCE agents use their agent desktop software to initiate the transfers. If the Unified CCE agents elect to use their IP phones for the transfer, the solution must be configured with CTI Route Points in Unified Communications Manager. This is because agent desktop software queries the ICM DNP for a label on where the call should be sent based on the number dialed by the agent. This is an important consideration, especially if the solution requires transfers to be initiated from both the IP Phone and the agent desktop software. The ladder diagrams provided in Figures 3-18 and 3-19 illustrate this interaction.

The details are included in these steps:

Step 1. The Unified CCE Agent initiates a transfer, either by dialing a route point configured on CUCM or dialing a number that has been configured in the ICM Dialed Number Plan. For the route point, because it is under the control of the Communications Manager PG, ICM is notified when the route point is called, which kicks off a routing script. For the ICM Dialed Number Plan, the agent desktop application sends the dialed digits via the Computer Telephony Integration Object Server (CTIOS) Server, which interacts with ICM without CUCM knowing yet that a transfer has been initiated.

Step 2. Unified ICM executes a transfer script associated with either the route point dialed or the result of the number mapping configured in the ICM Dialed Number Plan.

Step 3. The Unified ICM script executed in Step 2 either finishes by finding an available Unified CCE agent and returns the DN of that agents phone, or it returns a label that is then sent by CUCM to a Unified CVP Call Server to gain call control on the new call transfer leg.

Step 4. CUCM matches the label returned by Unified ICM in Step 3 and determines that it must be sent to the SIP Proxy Server via the SIP trunk configured for the CUCM cluster.

Step 5. The SIP Proxy Server consults its routing table and determines that the label dialed by CUCM in Step 4 must be sent to a Unified CVP Call Server and processed by its internal SIP Service.

Step 6. The Unified CVP Call Server's SIP service accepts the SIP INVITE from the SIP Proxy Server and hands the existing call request over to the ICM Service, which forwards it to the VRU PG.

Step 7. The Unified CVP Server sends Unified ICM an existing call request via its VRU PIM that is configured and hosted by the PG. This existing call request causes Unified ICM to continue the execution of the script that began in Step 2.

Note The label referenced in Step 6 and VRU labels used during the VRU leg of a call are two different labels because the routing client from Unified ICM's perspective during this transfer is actually the CUCM PG and not the Unified CVP VRU PG. A Routing Client is used to determine what label should be returned and to whom the label should go to. Later, this chapter discusses interaction with Unified ICM in much greater detail when VRU types and their significance are covered. The key takeaway is that a different routing client label is used during Steps 6 and 7, than the VRU label that will be returned in Steps 8 and 9. Also both of these labels have a correlation ID appended to the end of the label that was configured for the routing client. The Correlation ID is generated by ICM when it returns a label for the requesting routing client. Unified CVP uses its maximum DNIS setting within the call server to determine when the length of the received label signifies a new call or an existing call. For example, the default setting for the maximum DNIS in Unified CVP is ten digits. So for new calls originating on the switch leg, the total DNIS value of the incoming call should be ten digits or less, for Unified CVP to set up a new call dialog with Unified ICM. However, when a transfer or VRU label is provided to CVP as noted in the preceding Steps 6 and 7, the label for both routing clients, Unified CVP and CUCM, should be at least ten digits in length (that is, 9999999999 and 8888888888, respectively).

This ensures that when Unified ICM adds the call's Correlation ID to the routing client label, it will always be greater than ten digits and therefore always instructs Unified CVP to open an existing call dialog with Unified ICM to continue executing the transfer script started in Step 2. The maximum DNIS setting in Unified CVP can be modified to a lower or higher number, but it is critical to understand the relationship between this setting and which call dialog gets invoked based on the label presented to CVP.

Step 8. The Unified ICM routing script determines that the caller must be transferred to the VRU and passes a Connect to VRU request to the PG, which will be forwarded to the ICM Service on the Unified CVP Call Server. The new label generated here is for the Unified CVP Server because it is now the routing client. A single SendToVRU Node can be used to generate two labels: one from execution in Step 3 and one from its continued script execution in this step. Unified ICM is smart enough to determine which routing client label to return, and it accomplishes this with a single node in the script.

Step 9. VRU PG passes the information provided in Step 8 to the ICM Service on the Unified CVP Call Server.

Step 10. The Invitation to Connect to VRU goes from SIP Service back to the SIP Proxy Server requiring the SIP Proxy Server to determine which VXML gateway, based on its routing table, should handle the Connect to VRU and subsequent call treatment session for the VRU leg of the call.

Step 11. The Invitation (including information about the Ingress Gateway) goes from SIP Proxy Server to a VXML Gateway, which then connects the audio path back to the transferring the Unified CCE Agent's phone. The transferring Unified CCE Agents' Phone establishes audio connection with the VXML Gateway.

Step 12. The VRU label causes the VXML Gateway to fire off an application dial-peer on the VXML gateway, which starts the VRU or application leg of the call. The VXML Gateway connects to the Unified CVP Call Server via HTTP and requests instructions for treating the connected call. This HTTP new call request is handled by the Unified CVP Server's IVR Service, which then passes this request to the Unified CVP Server's ICM Service. Steps 11–18 (refer to Figure 3-12) continue to execute enabling the new transfer leg to either be treated on the VXML Gateway or connected to an agent in the new skill group.

Note IP initiated transfers from a CUCM are always a new call leg. This explains why a Unified CVP Call Server must be contacted during this transfer to enable its SIP Service to gain call control. A transfer leg is no different than a new switch leg, other than the DNIS called is treated as if it were an existing call from the ICM's perspective. The call control provided by the CVP on this new transfer call leg is completely different than the call control provided for the originating call leg that is being transferred.

SIP Call Flow Ladder Diagram

Figure 3-15 illustrates this model's call flow, which provides details on how comprehensive it truly is.

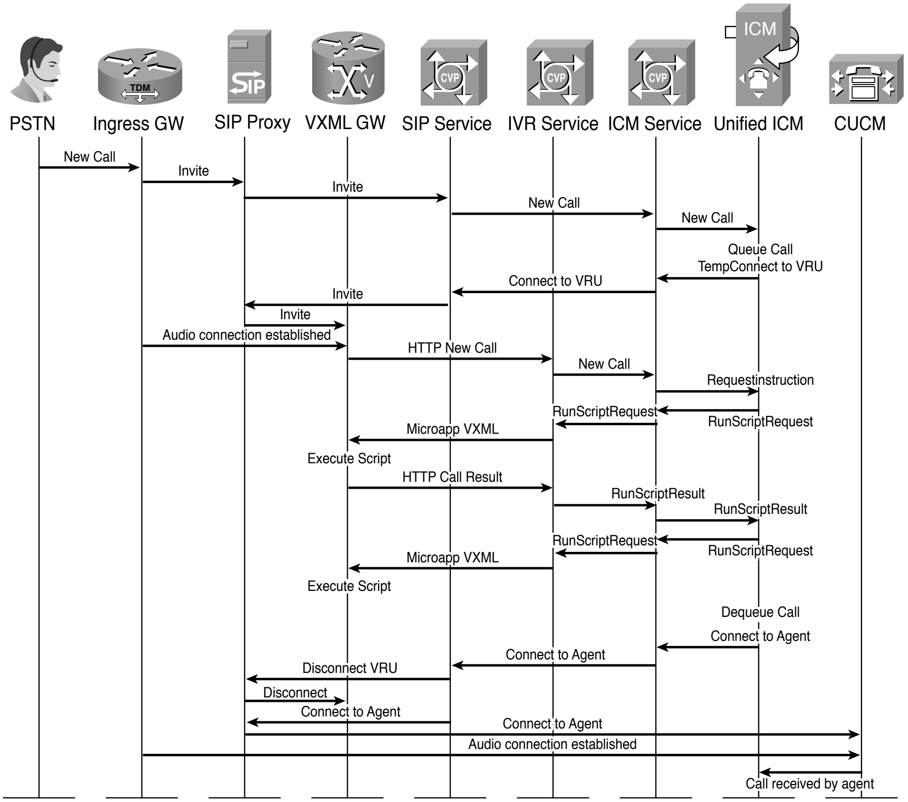


Figure 3-15 *Unified CVP Comprehensive SIP Call Flow Ladder Diagram*

In addition to this SIP call flow ending with the call being received by an agent, Figures 3-16 and 3-17 provide a ladder diagram for warm transfers using a CUCM route point to an available agent and to a queue treated by the VXML Gateway, respectively.

Dialing the CUCM route point initiates a connection to the CUCM PG that causes it to immediately execute a script within Unified ICM based on the dialed number of the route point (refer to Figures 3-16 and 3-17). The interaction when an agent is available and the call does not need to be queued to the VRU leg after the CVP has call control and has communicated with Unified ICM on what its next steps should be (refer to Figure 3-16).

Unified ICM discovers that a Unified CCE Agent is unavailable to take the call and instead makes a decision that the call should be queued and treated at the VRU (refer to Figure 3-17). Further communication is illustrated for the subsequent VRU or application leg of the call with conversations firing off between the VXML router and the Unified CVP Call Server.

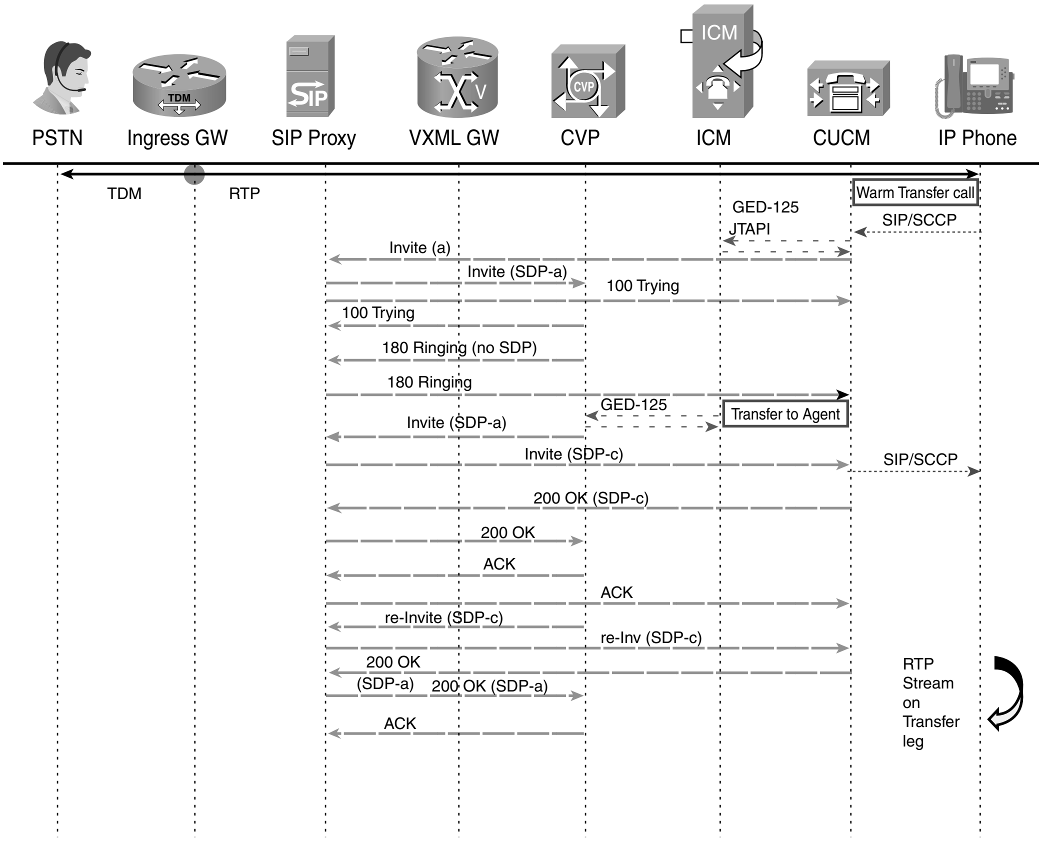


Figure 3-16 *Unified CVP Comprehensive SIP Warm Transfer to Agent via CUCM Route Point*

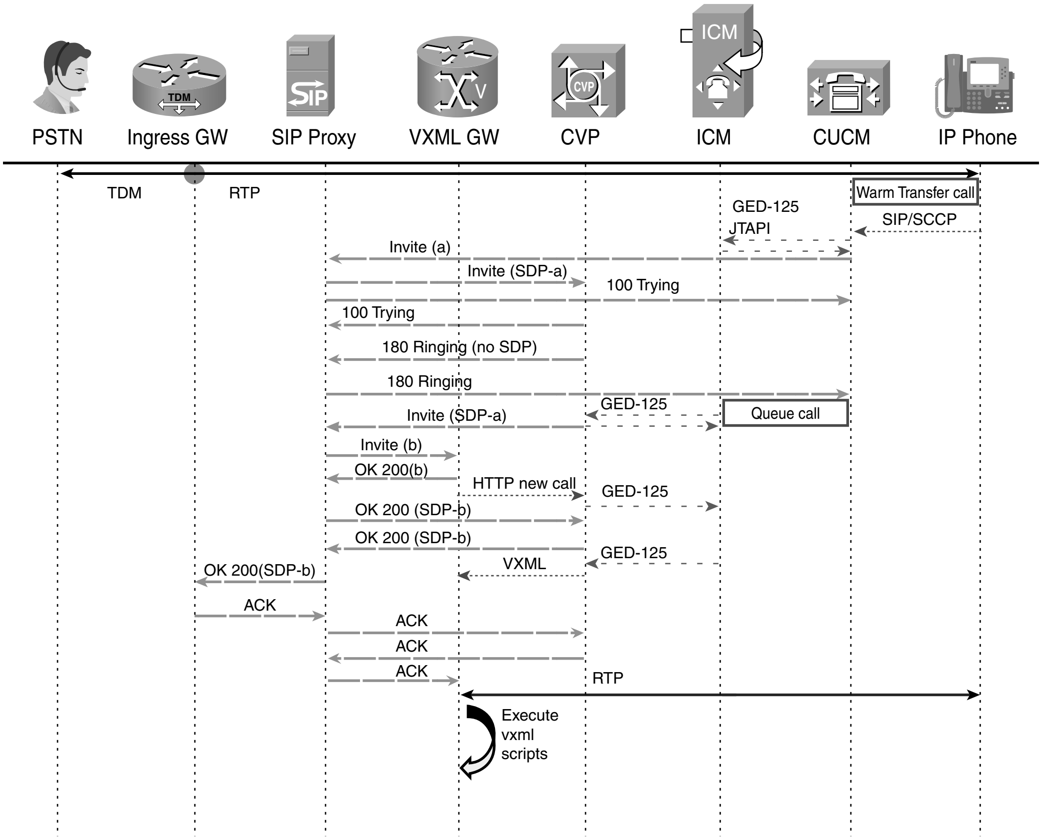


Figure 3-17 *Unified CVP Comprehensive SIP Warm Transfer to Queue via CUCM Route Point*

Transfers initiated by using ICM's Dialed Number Plan are similar except for how they are initiated. Figures 3-18 and 3-19 provide the equivalent ladder diagrams for transfer to an agent or a queue when initiating the transfers using an ICM Dialed Number Plan configuration.

The most important observation with respect to these ladder diagrams is how the transfer actually begins. When an Unified CCE Agent initiates the warm transfer via their agent desktop application, the ICM DNP application is engaged before CUCM must resolve and route the dialed number. ICM DNP enables the dialed number to translate within ICM and even have a script that executes the process of returning a completely different dialed number than the one the agent dialed. This is passed to CUCM for routing. The remainder of the ladder diagram displayed in Figures 3-18 and 3-19 are identical to the previous ones provided in Figures 3-16 and 3-17.

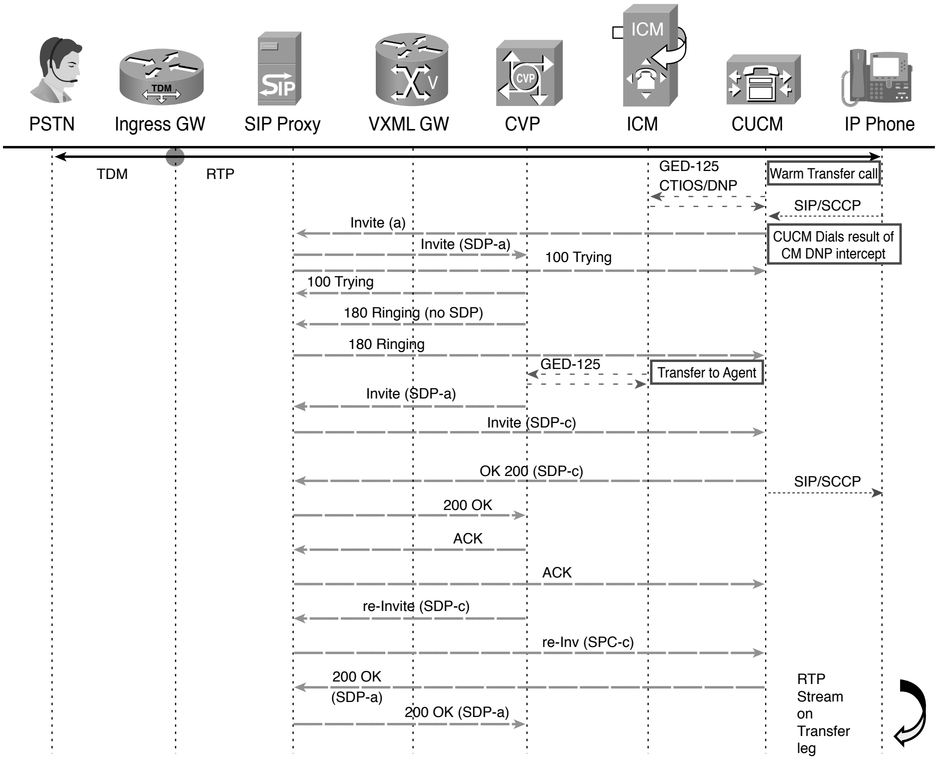


Figure 3-18 *Unified CVP Comprehensive SIP Warm Transfer to Agent via ICM Dialed Number Plan*

H.323 Protocol-Level and Component Call Flow

Figure 3-20 illustrates how the comprehensive call flow works when using H.323 as the call control protocol. The call closely resembles the previous SIP call flows presented in Figure 3-12. However, because H.323 is a peer-to-peer call control protocol, the gatekeeper provides only admission control services, which is different than how a SIP proxy handles the SIP invites in the SIP call flow. In the H.323 call flow for comprehensive, the gatekeeper replaces the SIP proxy from a call flow perspective. However, it is a mandatory component for H.323. In other words, there is no capability for Unified CVP to bypass the use of a gatekeeper as in the previously illustrated SIP without proxy call flow. Even if a Set transfer label configuration is used with H.323, enabling calls to be sent to the originating gateway during the switch leg of the H.323 call flow, the H.323 service does not go into an UP state unless a gatekeeper is configured. Without the H.323 service active on the call server, H.323 connections cannot be made during the switch leg of a call or connected to a VXML gateway after the VRU leg of the call is engaged. This creates a classic "chicken before the egg" stalemate situation.

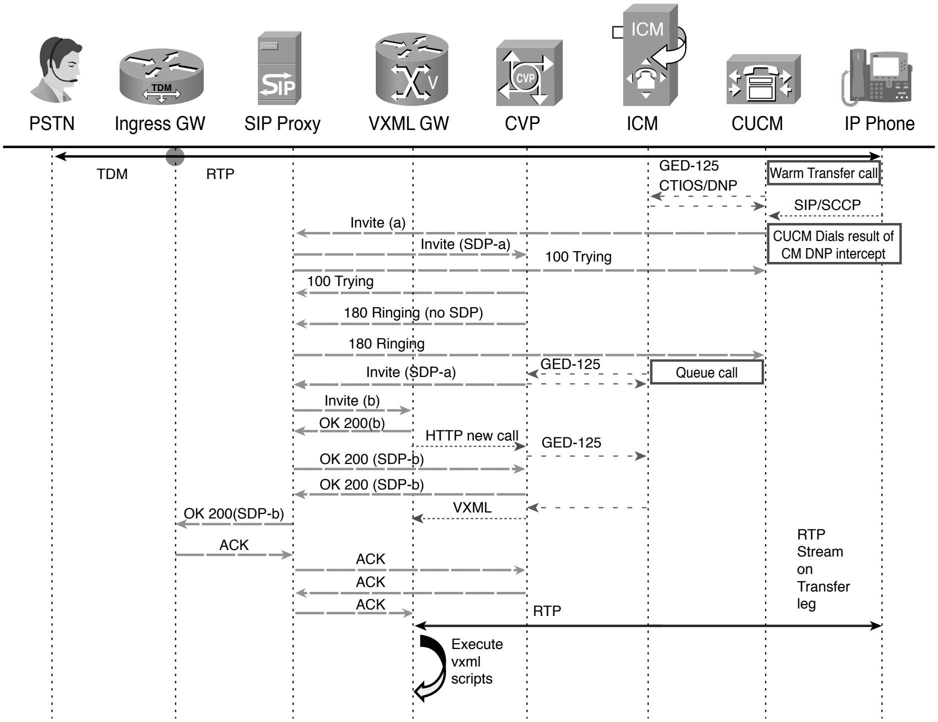


Figure 3-19 *Unified CVP Comprehensive SIP Warm Transfer to Queue via ICM Dialed Number Plan*

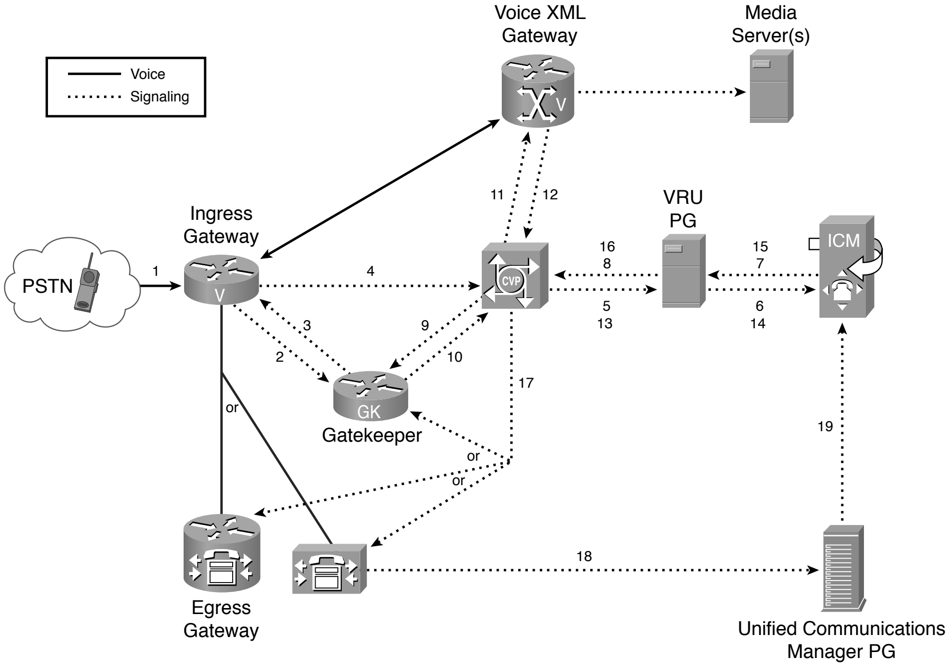


Figure 3-20 *Unified CVP Comprehensive H.323 Call Flow*

Following are the details for the steps previously referenced:

Step 1. The call arrives from either the PSTN or a VoIP connection to the gateway.

Step 2. The Ingress Gateway sends a Registration, Admission, and Status (RAS) request to the H.323 Gatekeeper to find the IP address of the Unified CVP Call Server.

Step 3. The H.323 Gatekeeper executes its call routing decision tree and matches the E.164 number (DNIS) to a registered Unified CVP Call Server. The gatekeeper returns an Admission Confirm (ACF) message containing the IP address of the Unified CVP Call Server to the Ingress Gateway.

Step 4. The Ingress Voice Gateway sends a H.225 call setup message to the Unified CVP Server's H.323 Service.

Step 5. The Unified CVP Server sends Unified ICM a new call request via its VRU PIM configured and hosted by the PG.

Step 6. This new call request invokes a new incoming dialed number that in turn invokes a routing script in Unified ICM.

Step 7. The Unified ICM routing script determines that the caller must be transferred to the VRU and passes a Connect-to-VRU request to the PG, which will be forwarded to the ICM Service on the Unified CVP Call Server.

Step 8. PG passes the information provided in Step 6 to the ICM Service on the Unified CVP Call Server.

Step 9. 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 VRU label returned by Unified ICM.

Step 10. The H.323 Gatekeeper executes its call routing decision tree and matches the VRU label to a registered VoiceXML Gateway. The gatekeeper returns an Admission Confirm (ACF) message containing the IP address of the VoiceXML Gateway to the Unified CVP Call Server.

Step 11. The Unified CVP Server's H.323 Service sends an H.225 setup message to the VoiceVXML Gateway returned in Step 10 by the Gatekeeper's ACF message.

Step 12. The VRU label causes the VXML Gateway to fire off an application dial-peer on the VXML Gateway, which starts the VRU or application leg of the call. The VXML Gateway connects to the Unified CVP Call Server via HTTP and requests instructions for treating the connected call. This HTTP new call request is handled by the Unified CVP Server's IVR Service, which then passes this request to the Unified CVP Server's ICM Service.

Step 13. ICM Service sends Request Instructions to Unified ICM via the PG.

Step 14. Unified ICM continues the script that was started in Step 7. It processes additional nodes that produce Run Script requests to the Unified CVP Server's

ICM Service, which then hands these instructions over to the Unified CVP Server's IVR Service that converts them into VXML pages forwarded back to the VXML Gateway for rendering and execution. This process can also include prompt and collect instructions and continues between Unified ICM, Unified CVP, and the VXML gateway until an agent becomes available. The VoiceXML Gateway also fetches any media files requested in the VXML pages from the media server (refer to the top of Figure 3-20).

Step 15. An agent becomes available, so Unified ICM dequeues the call and asks to be disconnected from the VXML Gateway. Unified ICM passes a Connect-to-Agent request to the Unified CVP Server's ICM Service via the VRU PG.

Step 16. PG passes the information provided in Step 15 to the ICM Service on the Unified CVP Call Server. The ICM Service passes this connect request to the Unified CVP Server's IVR Service.

Step 17. The Unified CVP Server's IVR Service requests the H.323 Service to transfer the caller to the dialed number of the selected agent or Egress Gateway. The H.323 service then sends a RAS message to the H.323 Gatekeeper to find the desired endpoint, either an Egress Gateway or a H.323 CUCM trunk. The gatekeeper then returns an ACF message containing the IP address of the termination point. This causes the Unified CVP Server's H.323 service to send a H.225 call setup message to the termination point. Because the audio path was torn down between the Ingress Gateway and the VXML Gateway, a new one is established between the Ingress Gateway and an Egress Gateway or a Unified CCE Agent IP Phone hosted on the CUCM cluster.

Step 18. The Unified Communications Manager notifies its Call Manager PG that the agent has received the call.

Step 19. The Call Manager PG informs Unified ICM that the call was received by the agent.

VoIP Transfers Using H.323

As discussed earlier with VoIP transfers using SIP, H.323 implements the same call flow when using ICM DNP or CUCM Route Points. The only difference in the two call flows is the use of a SIP Proxy for SIP and a gatekeeper for H.323 transfers. In addition, the CUCM server that sets up the transfer call leg also uses a H.323 trunk connected to a gatekeeper versus a SIP trunk connected to a SIP Proxy. Other than these two small differences, the call flow is essentially the same. The SIP INVITE messages are replaced with H.323 ACF messages enabling the Unified CCE agents to connect to either a VoiceXML Gateway for treatment, an Egress Gateway connected to TDM endpoints, or an H.323 CUCM trunk passing the transfer call to an available agent handling a different Unified ICM skill group. The fundamentals that enable Unified CVP to gain call control for the transfer leg exists both in SIP and H.323 transfers. Only the component that holds the dial plan and decision on where the call is transferred changes, in this case from a SIP Proxy to a H.323 Gatekeeper. Figure 3-21 outlines a VoIP transfer when using H.323.

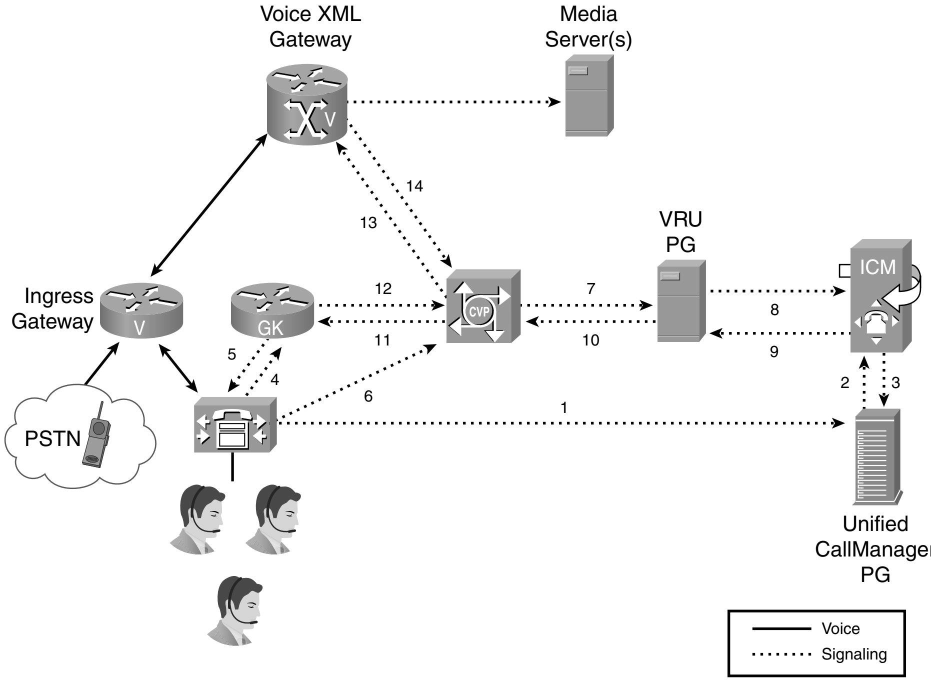


Figure 3-21 *Unified CVP Comprehensive H.323 Transfer Call Flow Using ICM DNP*

Following are the details for the steps referenced in Figure 3-21:

Step 1. The Unified CCE Agent initiates a transfer, either by dialing a route point configured on CUCM or dialing a number that has been configured in the ICM Dialed Number Plan. For the route point, because it is under the control of the Communications Manager PG, ICM is notified when the route point is called, which invokes a routing script. For the ICM Dialed Number Plan, the agent desktop application sends the dialed digits via the CTIOS server, which interacts with ICM without CUCM knowing yet that a transfer has been initiated.

Step 2. Unified ICM executes a transfer script associated with either the route point dialed or the result of the number mapping configured in the ICM Dialed Number Plan.

Step 3. The Unified ICM script executed in Step 2 either finishes by finding an available Unified CCE agent and returns the label of the agents extension, or it returns a label that must then be sent by CUCM to a Unified CVP Call Server to gain call control on the new call transfer leg.

Step 4. CUCM matches the label returned by Unified ICM in Step 3 and determines that it must be checked against the gatekeeper accessible via a RAS message over an H.323 trunk configured for the CUCM cluster.

Step 5. The H.323 Gatekeeper executes its call routing decision tree and matches the E.164 number (DNIS) to a registered CVP Call Server and processed by its internal H.323 Service. The H.323 Gatekeeper returns an ACF message to the CUCM subscriber processing the H.323 trunk information, which contains the IP address of the Unified CVP Call Server.

Step 6. CUCM sends a H.225 setup message with the Unified CVP Call Server's H.323 service.

Step 7. The Unified CVP Call Server's H.323 service hands the existing call request over to the ICM Service, which forwards it to the PG.

Step 8. The Unified CVP Server sends Unified ICM an existing call request via its VRU PIM configured and hosted by the PG. This existing call request causes Unified ICM to continue the execution of the script started in Step 2.

Step 9. The Unified ICM routing script determines that the caller must be transferred to the VRU and passes a Connect-to-VRU request to the PG. This is forwarded to the ICM Service on the Unified CVP Call Server. The new label generated here is for the Unified CVP Server because it is now the routing client. You can use a single SendToVRU Node to generate two labels: one from execution in Step 3 and one from its continued script execution in this step. Unified ICM is smart enough to determine which routing client label to return, and it accomplishes this with a single node in the script.

Step 10. PG passes the information provided in Step 9 to the ICM Service on the Unified CVP Call Server.

Step 11. The message to connect to VRU travels from the H.323 Service back to the H.323 Gatekeeper. Based on its prefix table, it requires the gatekeeper to determine which VoiceXML Gateway should handle the Connect to VRU and subsequent call treatment session for the VRU leg of the call.

Step 12. The Unified CVP Call Server's H.323 Service sends an H.225 setup message with the VoiceXML Gateway, which then connects the audio path back to the transferring Unified CCE Agent's phone. The transferring Unified CCE Agents' phone establishes audio connection with the VXML Gateway.

Step 13. An ACF message is returned to the Unified CVP Call Server's H.323 service that contains the IP address of the VoiceXML Gateway that should handle the VRU leg of the call.

Step 14. The VRU label causes the VXML Gateway to fire off an application dial-peer on the VXML Gateway, which starts the VRU or application leg of the call. The VXML Gateway connects to the Unified CVP Call Server via HTTP and requests instructions for treating the connected call. This HTTP new call request is handled by the Unified CVP Server's IVR Service, which then passes this request to the Unified CVP Server's ICM Service. As shown in Figure 3-20, Steps 13 through 19 continue to execute enabling the new transfer leg to either be treated on the VXML Gateway or connected to an agent in the new skill group.

H.323 Call Flow Ladder Diagram

Figure 3-22 illustrates a call flow ladder diagram when the H.323 protocol is engaged by a Unified CVP comprehensive deployment. This ladder diagram is similar to the diagram presented in Figure 3-15, with the SIP Proxy server replaced by an H.323 Gatekeeper. As noted in the previous section, H.323 transfers using ICM DNP or a CUCM Route Point are exactly the same as they are when using SIP, with the exception of the use of an H.323 trunk and a gatekeeper for admission control.

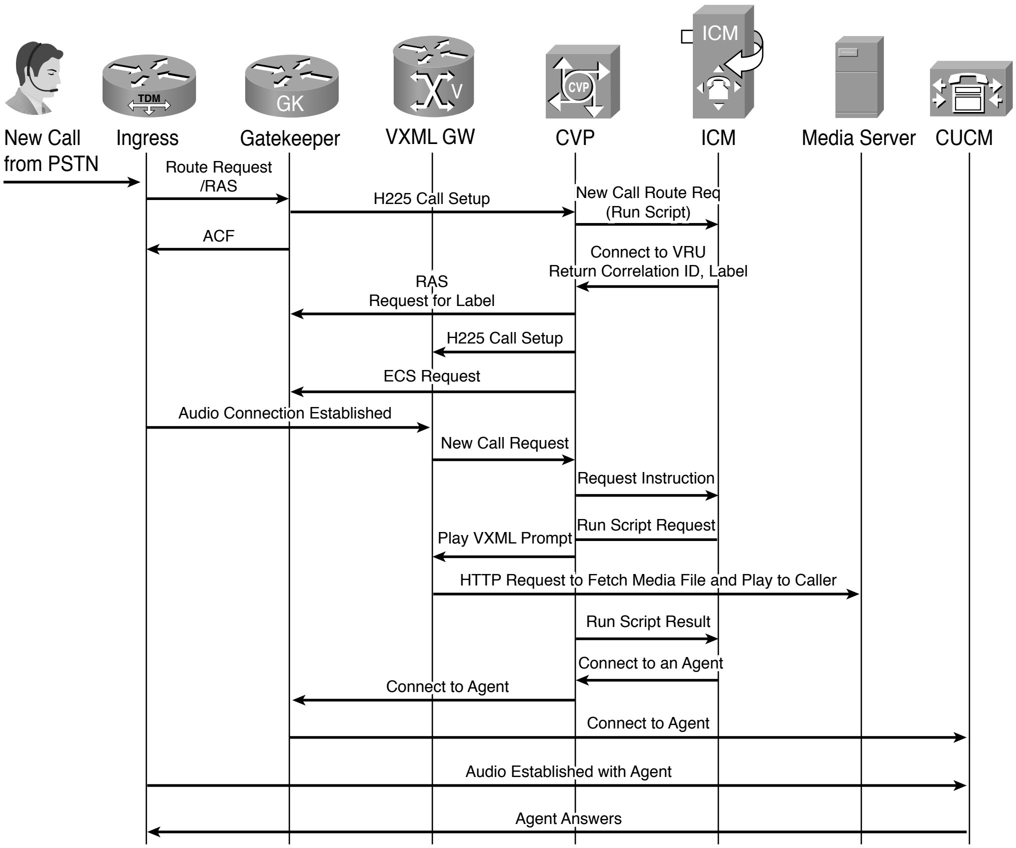


Figure 3-22 *Unified CVP Comprehensive H.323 Call Flow Ladder Diagram*

Figures 3-23 and 3-24 provide a ladder diagram for warm transfers using a CUCM route point to an available agent and to a queue treated by the VXML Gateway, respectively.

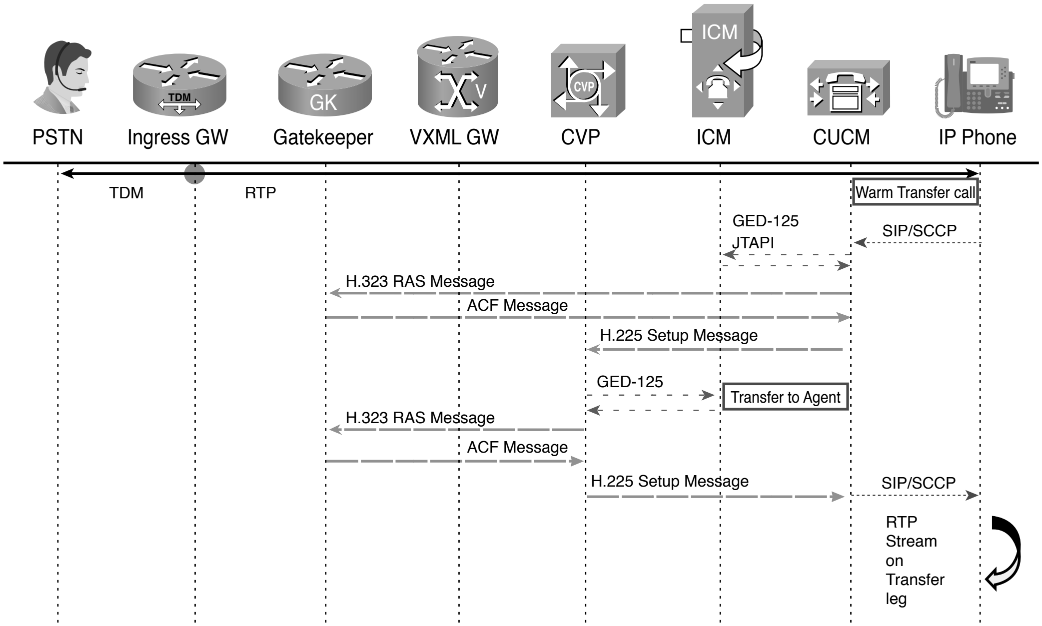


Figure 3-23 *Unified CVP Comprehensive H.323 Warm Transfer to an Agent via CUCM Route Point*

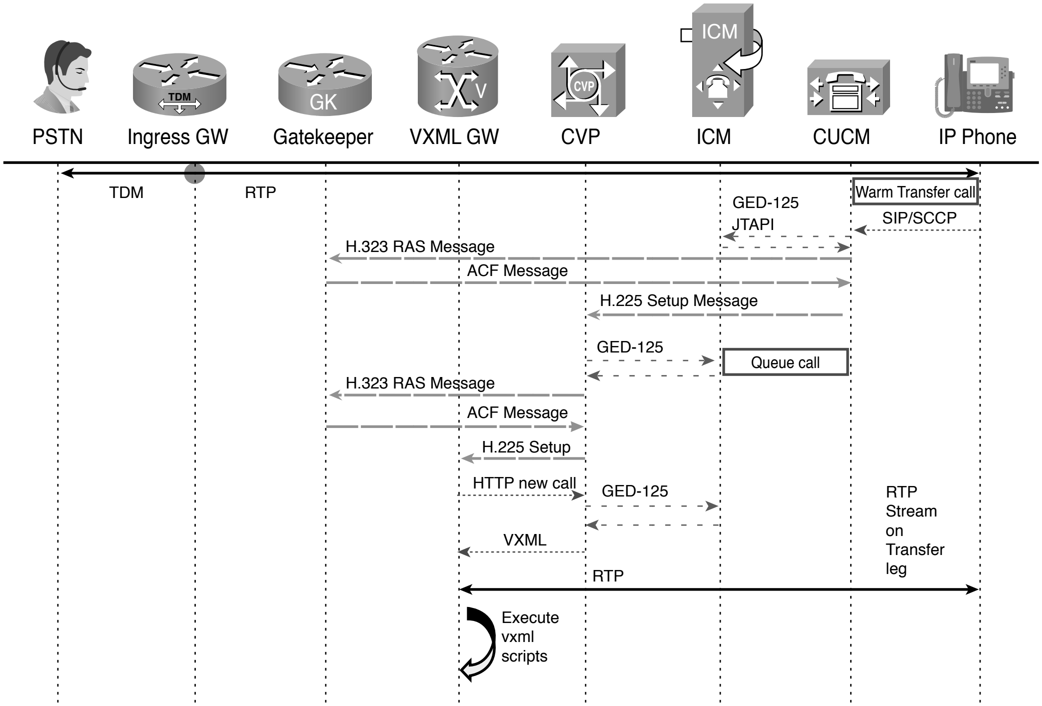


Figure 3-24 *Unified CVP Comprehensive H.323 Warm Transfer to Queue via CUCM Route Point*

All the observations noted for similar ladder diagrams found in Figures 3-16 and 3-17 for SIP-based transfers also apply here. One additional but equally significant observation with H.323 is how the H.323 Gatekeeper is never involved in call setup or control, only providing registration, access, and status services. Because H.323 is a peer-to-peer call control protocol, when the ACF messages are received, the endpoints set up calls between each other. They do not require a proxy or broker to complete on their behalf.

For consistency, Figures 3-25 and 3-26 provide the equivalent ladder diagrams for transfer to an agent or a queue when initiating the transfers using an ICM Dialed Number Plan configuration and H.323.

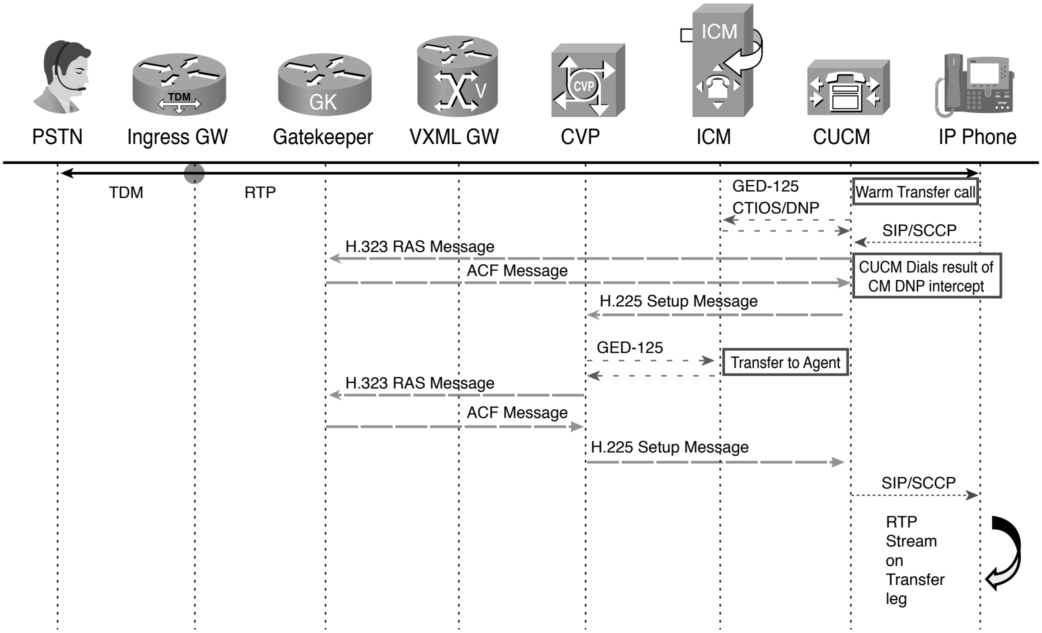


Figure 3-25 *Unified CVP Comprehensive H.323 Warm Transfer to an Agent via ICM DNP*

As in SIP transfers that use ICM DNP, the transfer starts and ends the same way. The usage of the gatekeeper (refer to Figures 3-22 and 3-23) is also identical when employing this type of transfer.

Transfers and Subsequent Call Control

In addition to the transfers managed by Unified ICM, the Comprehensive Deployment model can transfer calls to non-ICM terminations or invoke a Release Trunk Transfer to the PSTN. However, if a call is transferred to a non-ICM controlled termination, call data cannot be passed to the termination, further call control is impossible for the call, and the cradle-to-grave call reporting that Unified ICM gathers is complete. For a Release to Trunk

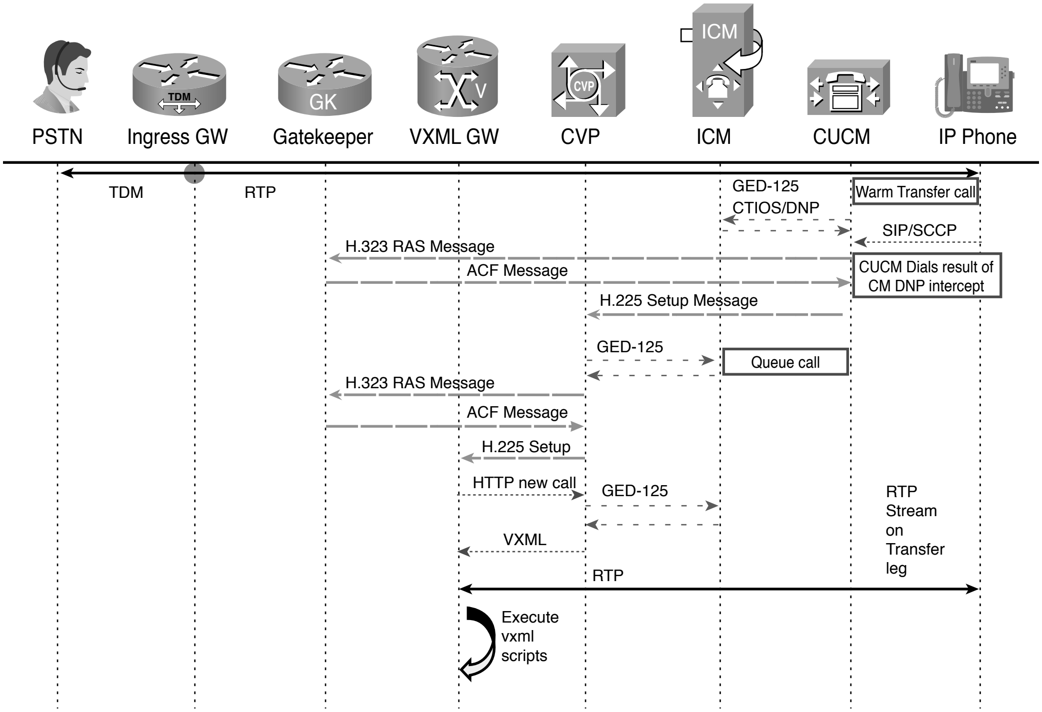


Figure 3-26 *Unified CVP Comprehensive H.323 Warm Transfer to Queue via ICM DNP*

Transfer on the Ingress Voice Gateway, call data or call control cannot be maintained. However, if the call is translation routed to another ICM peripheral, call data and cradle-to-grave reporting can be maintained.

If a transfer fails or the termination device returns a 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 within Unified ICM, enabling a different target to be selected or execution of a remedial action.[4]

The VRU-Only Model

The last functional deployment model for Unified CVP exists for organizations that use advanced PSTN switching services controlled via a Cisco Unified ICM PSTN Network Interface Controller (NIC). There are two Unified ICM PSTN NICs available that enable subsequent call control of calls in the PSTN: the SS7 NIC and the Carrier Routing Service Protocol (CRSP) NIC. These NICs provide Unified ICM the capability to preroute calls intelligently to Unified ICM peripherals (such as ACDs and IVRs) and perform mid-call transfers in the PSTN.[4] Figure 3-27 shows the components used with this solution and their protocols.

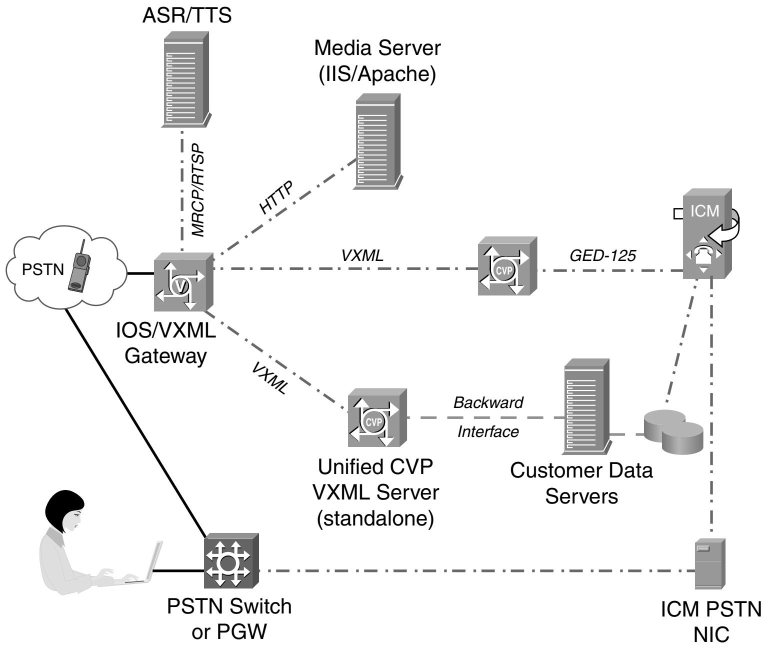


Figure 3-27 *Unified CVP VRU-Only Functional Deployment Model*

Table 3-4 identifies which components are required, optional, and not used by this model. In addition, the Native column identifies components that are native to the Unified CVP solution.

Table 3-4 *Unified CVP VRU-Only Native and Non-Native Component Usage*

Component	Required	Optional	Not Used	Native
SIP Service (Call Server)	—	—	Yes	Yes
IVR Service (Call Server)	Yes	—		Yes
ICM Service (Call Server)	Yes	—		Yes
H323 Service (Call Server)	—	—	Yes	Yes
VoiceXML Server	—	Yes	—	Yes
Unified Call Studio	—	Yes	—	Yes
Ingress Gateway	—	Yes	—	—
VXML Gateway	Yes	—	—	—
SIP Proxy	—	Yes	—	—
Gatekeeper	—	Yes	—	—

Table 3-4 *Unified CVP VRU-Only Native and Non-Native Component Usage*

Component	Required	Optional	Not Used	Native
Operations Console	Yes	—	—	Yes
Reporting Server	—	Yes	—	Yes
ASR/TTS	—	Yes	—	—
Media Server	—	Yes	—	—
DNS Server	—	—	Yes	—
Content Services Switch	—	Yes	—	—
Unified ICM	Yes	—	—	—
Unified Call Manager	—	Yes	—	—
Egress Gateway	—	Yes	—	—

Although Table 3-4 lists the Unified Communications Manager and Egress Gateway as optional components, typically at least one of these components exists to enable a call to be transferred to either a legacy ACD agent via an Egress Gateway or to a Unified CCE Agent hosted by a Unified Communications Manager cluster. The component to be deployed depends entirely on the organization's requirements. In addition, the use load balancers, media servers, and even an ASR/TTS Server are also optional because the Unified CVP VXML Gateways can be configured in such a way to enable them to operate without these components. For ASR/TTS, this would be true only if the organization does not have requirements to perform ASR/TTS treatment for incoming calls. If the IOS Ingress Gateway does not perform VoiceXML Gateway duties, a SIP Proxy or H.323 Gatekeeper can be used to route the VRU label to an available VoiceXML Gateway. The most important takeaway for this model is that Unified CVP does not perform call control for the switch leg of the call. The PSTN Switch or a Cisco Packet Data Network Gateway PGW with Unified ICM handles all call control and prerouting for the switch leg of the call leaving Unified CVP to handle only the VRU leg of the call—therefore the name for this model. Unified ICM can pass call data between termination points (for screen pop or other intelligent treatment) and provide cradle-to-grave reporting for all calls.

Component Call Flow

This section details the call flow steps performed in a typical VRU-Only implementation. As the name indicates, this call flow focuses on the VRU leg of the call. It leaves the call control and switch leg the responsibility of the PSTN's carrier switch and Unified ICM. Figure 3-28 illustrates this call flow with an optional SIP Proxy injected to load balance the VRU leg of the call.

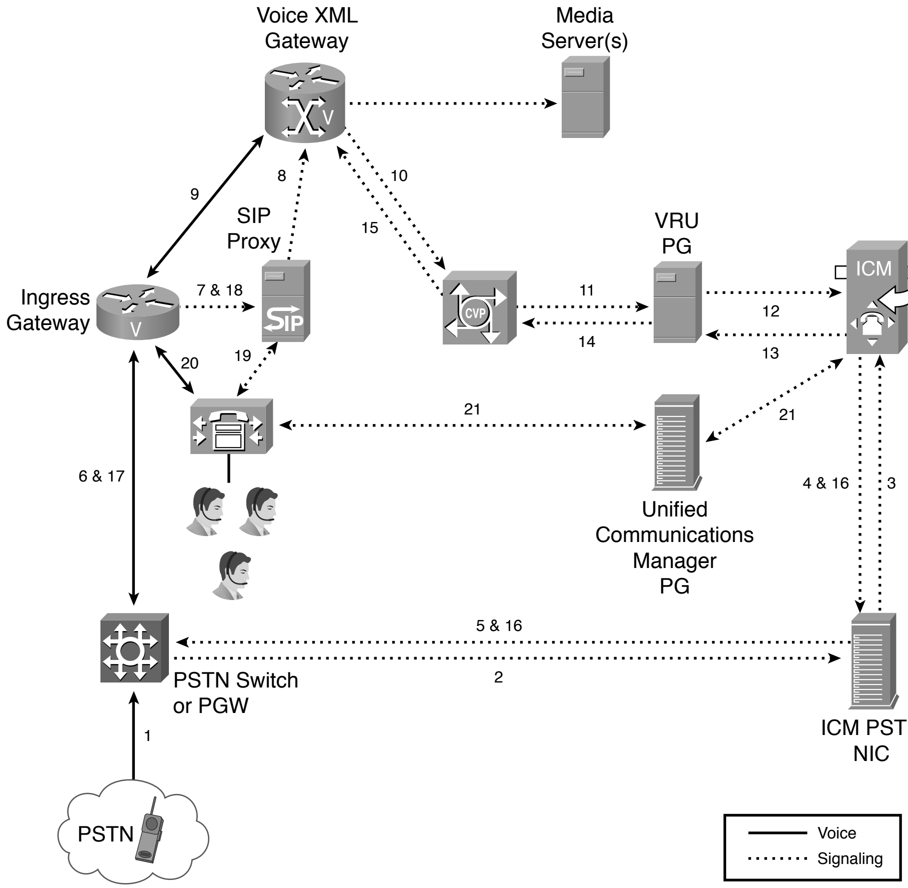


Figure 3-28 *Unified CVP VRU-Only Call Flow*

Following are the details for the steps referenced in Figure 3-28:

Step 1. The call arrives at the PSTN Switch or PGW.

Step 2. The PSTN Switch sends a new-call message to Unified ICM via either a CRSP NIC or a SS7 NIC.

Step 3. The NIC forwards the request to Unified ICM where a routing script is invoked based on the dialed number provided by the PSTN Switch.

Step 4. This routing script uses either a Send to VRU node or a Translation Route to VRU node to send a result back to the ICM PSTN NIC.

Step 5. The NIC sends the result back to the PSTN Switch to have the call routed to the Unified CVP Ingress Voice Gateway. Depending on the PSTN capability and Unified ICM VRU 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 a correlation ID.

Note In most legacy ACD and IVR cases, translational routing is used to enable the call data to survive when the PSTN switch connects to these peripherals for further instructions from Unified ICM. These peripherals pass these translation route labels to their respective Unified ICM PGs, which then "marry" the call up with ICM for further processing and the extraction of call data. For Unified CVP, both translational routing and the use of a dialed number plus a correlation ID is supported. The one used depends on how the VRU type is defined. However, if translational routing is wanted or required, the translational route label ranges must be configured on all Unified CVP Call Servers that will be processing these labels. This is done via the Unified CVP Call Server's ICM Service using the Operations Console. Failure to correctly build these label ranges results in failed calls during the VRU leg of the call.

Step 6. The PSTN routes the call to an available Ingress Voice Gateway port. At this point in the call flow, the VRU leg is beginning because the switch leg was completed when Step 5 was executed.

Step 7. The Ingress Voice Gateway performs normal inbound POTS dial-peer matching to deliver the call to an available VoiceXML Gateway port. It is at this point that the use of a SIP Proxy or H.323 Gatekeeper can be used to aid in the load balancing and routing of the call. Figure 3-28 illustrates the use of a SIP Proxy. It also assumes that the Ingress Gateway and VoiceXML Gateway are separate devices.

Step 8. The SIP Proxy extends an INVITE message to the VoiceXML Gateway to terminate the call between it and the Ingress Gateway.

Step 9. The SIP INVITE is forwarded from the SIP Proxy Server to a VXML Gateway, which then connects the audio path back to the Ingress Gateway.

Step 10. The VRU label causes the VXML Gateway to fire off an application dial-peer on the VXML Gateway, which starts the VRU or application leg of the call. The VXML Gateway connects to the Unified CVP Call Server via HTTP and requests instructions for treating the connected call. This HTTP new call request is handled by the Unified CVP Server's IVR Service, which then passes this request to the Unified CVP Server's ICM Service.

Note The VRU-Only model at this point in the call flow supports the use of content load balancers such as ACE or CSS to load balance this HTTP request to a Unified CVP Call Server. Without the use of a content load balancer, the VoiceXML Gateway can have up to two Unified CVP call servers configured. This is accomplished by configuring two parameters for the application on the gateway, param cvpserverhost and param cvpserverhostbackup. If a content load balancing pair is used, these parameters would reference the VIP for the primary and backup load balancers, respectively. In other models like comprehensive, the default behavior for the VoiceXML Gateway is to extract the call server information from the signaling and use that information to connect its VRU leg back to the same call server that processed the switch leg of the call. However, in a VRU-Only deployment, there is no switch leg handled by a Unified CVP call server setting this signaling information to null, causing the VoiceXML Gateway to use the configured application parameters. This is a requirement in VRU-Only deployments, and if this configuration step is omitted from VoiceXML Gateway configurations, calls fail during the start of the VRU leg on the VoiceXML Gateway.

Step 11. The Unified CVP Call Server's ICM Service sends a Request Instructions message to the VRU PG.

Step 12. The Unified CVP Server sends the Unified ICM a new call request via its VRU PIM configured and hosted by the PG.

Step 13. Unified ICM continues the script that was initiated in Step 3 and processes additional nodes that produce Run Script requests to the Unified CVP Server's ICM Service via the VRU PG.

Step 14. The VRU PG provides Run Script requests produced in Step 13 to the ICM Service located on the Unified CVP call server.

Step 15. The ICM Service hands these instructions over to the Unified CVP Server's IVR Service that converts them into VXML pages that are forwarded back to the VXML Gateway for rendering and execution. This process can also include prompt and collect instructions and continues between the Unified ICM, Unified CVP, and VXML Gateway until an agent becomes available. The VoiceXML Gateway also fetches any media files requested in the VXML pages from the media server (refer to the top of Figure 3-28). Steps 10 through 15 continue until the call is handled or needs to be transferred to an agent or other termination point.

Note The use of a VXML Server and ASR/TTS services can also be invoked by the VoiceXML Gateway at this point in the VRU leg of the call.

Step 16. 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. This connect message contains either a translation route label or a dialed number plus correlation ID (depending on the PSTN switches capabilities).

Step 17. Upon receipt of the connect message, the PSTN releases the existing call leg with Ingress Gateway and connects the call to the new termination point. In this example call flow, the same Ingress Gateway is used to connect the call to a Unified CCE Agent, so the PSTN connects back to the Ingress Gateway with a different dialed number.

Step 18. The Ingress Voice Gateway performs normal inbound POTS dial-peer matching to deliver the call to an available Unified Communications Manager SIP trunk. It is at this point that the use of a SIP Proxy or H.323 Gatekeeper can be used to aid in the load balancing and routing of the call. Figure 3-28 illustrates the use of a SIP Proxy. It also requires that a SIP trunk be configured on the CUCM cluster pointing at the SIP Proxy server. If a H.323 Gatekeeper is used instead, a H.323 trunk must be configured and registered between the gatekeeper and the CUCM cluster.

Step 19. The SIP Proxy forwards an INVITE message to the CUCM SIP trunk to terminate the call between the SIP trunk and the Ingress Gateway.

Step 20. The Ingress Gateway sets up an audio path with the Unified CCE Agent.

Step 21. CUCM notifies Unified ICM that an agent has received the call.

VoIP Transfers

VoIP transfers in this model are quite similar to the Comprehensive model with the exception of the component that has call control. In the Comprehensive model, Unified CVP has call control, and as illustrated earlier in this chapter, Unified ICM instructs Unified CVP to move the call to a new termination point. However, in the VRU-Only model, the PSTN has call control, so Unified ICM instructs the PSTN via the NIC to move the call to the new termination point. The previous call flow illustrated this beginning at Step 16, when a Unified CCE Agent became available and the call needed to be delivered to that agent. In addition, if a transfer is initiated by a Unified CCE Agent to another Unified CCE Agent on the same CUCM cluster, Unified ICM instructs Unified CM via its PG to transfer the call. As shown in the comprehensive model, this warm transfer leg is a new call leg, and the PSTN does not have call control scope over that type of transfer.

Call Flow Ladder Diagram

Figure 3-29 illustrates a call flow ladder diagram without the use of a SIP Proxy or gatekeeper. The PSTN connects directly to the VoiceXML Gateway, which kicks off the VRU leg of the call.

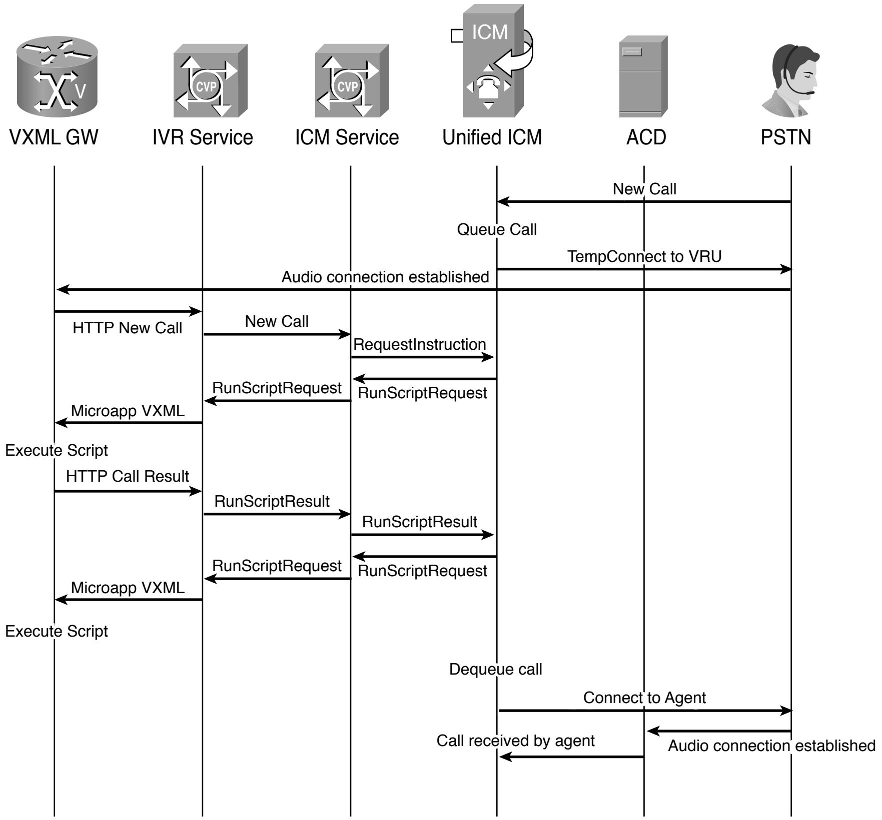


Figure 3-29 *Unified CVP VRU-Only Call Flow Ladder Diagram*

Network VRU Types

This section examines the different Network VRU types defined within Unified ICM and how they each relate to Unified CVP deployments. It begins with an overview of Unified VRU Types and then details how Unified CVP operates as a Type 10, 5, 3 and 7, or 8 and 2. The terms Voice Response Unit (VRU) and Interactive Voice Response (IVR) are used interchangeably in the following sections.

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 independently provide the Switch and VRU portions.

The call delivery to a 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 must correlate the two legs (Switch and VRU) of the same call to provide instructions for completing the call. In the Unified ICM application, the VRU must 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. This 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) must be configured in Unified ICM to reach the VRU. 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.

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.[5]

Unified CVP as a Type 10 VRU

Figure 3-30 shows the relationship between the switch and VRU leg of a call when using a Type 10 VRU.

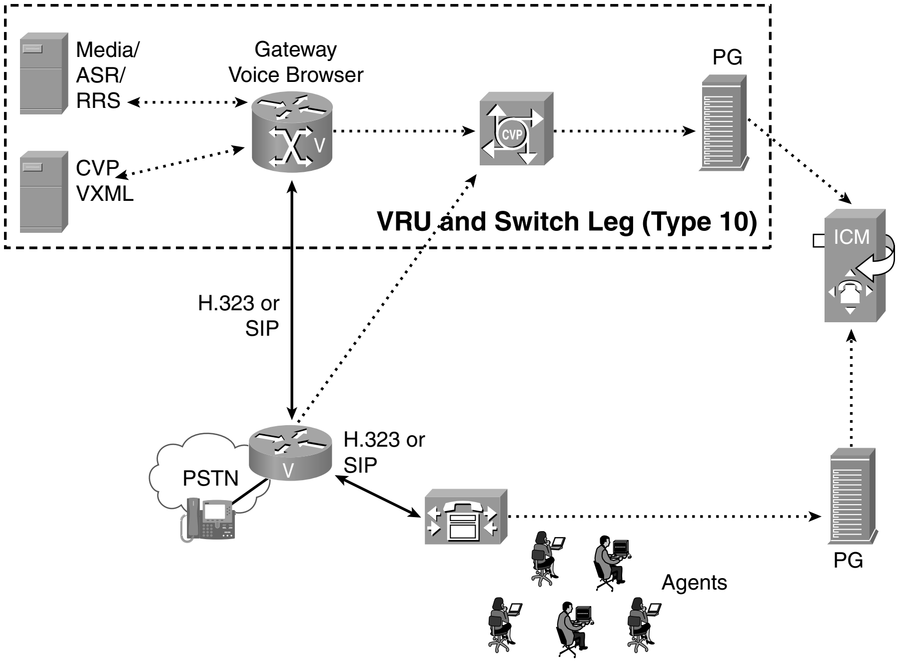


Figure 3-30 *Unified CVP as a Type 10 VRU*

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.

- 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.
- For calls originated by Cisco Unified Communications Manager, 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 transfers the call to the Unified CVP VRU leg. If calls will be transferred to Unified CVP from CUCM, it also needs another label for the CUCM routing client. That label transfers the call to the Unified CVP Switch leg. The Unified ICM Router sends that label to CUCM with a Correlation ID concatenated to it. CUCM 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 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.[6]

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 next). 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.

Unified CVP as Type 5 VRU

Types 5 and 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 they connect to the VRU. In Type 6, the Switch and the VRU are the same device; therefore, the call is already at the VRU. There is no need for a Connect and Request message sequence from Unified ICM's perspective. Figure 3-31 illustrates a Type 5 VRU with a Type 7.

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. 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 As noted in Chapter 2, "Unified CVP Architecture Overview," there are two legs of the call as perceived by Unified ICM: the Switch leg and the VRU leg. Where Unified CVP acts as the service node application (that is, when Unified CVP receives the call directly from the network and not via pre-routing as in the VRU-Only model) Unified CVP appears to Unified ICM as Type 5 because the call control (Unified CVP) and the VRU devices 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 configuration depending on the deployment model (for example, the VRU leg could be Type 7 in the Comprehensive Unified ICM Enterprise deployment model as illustrated in Figure 3-31). However, the preferred VRU type for all new implementations of Unified CVP is Type 10.

For configuration examples of the Unified CVP application with VRU Type 3 or Type 7, refer to the latest version of the Cisco Unified CVP Configuration and Administration Guide, available on Cisco.com.

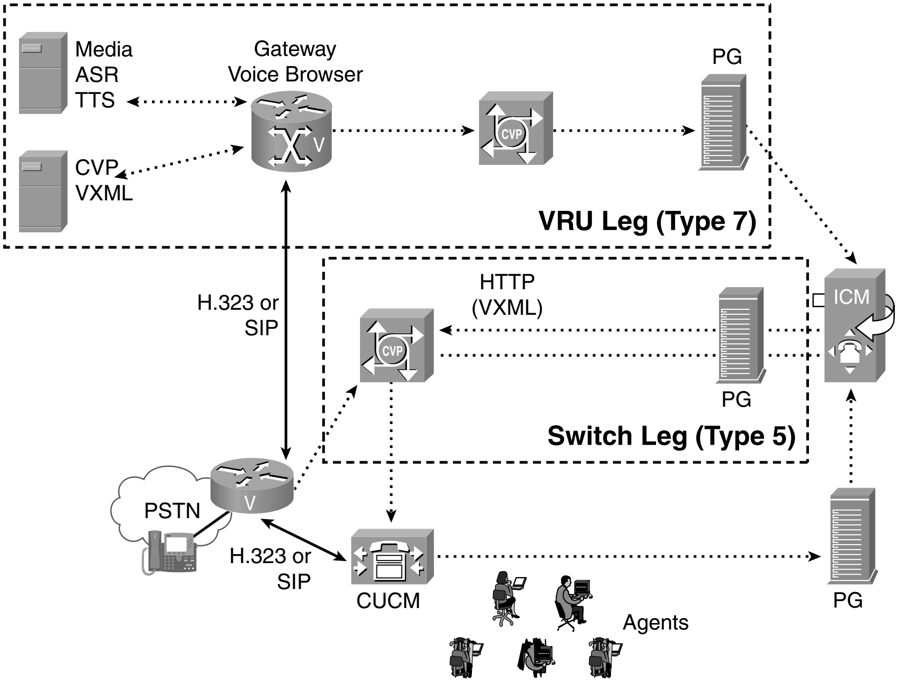


Figure 3-31 *Unified CVP as a Type 5 VRU*[6]

Neither Correlation ID nor Translation Route ID is needed when Unified CVP acts as a Type 5 VRU to Unified ICM and the NAM.[7]

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

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. 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)[8]

For configuration examples of the Unified CVP application with VRU Type 3 or Type 7, refer to the latest version of the Cisco Unified CVP Configuration and Administration Guide, available at Cisco.com.

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

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 capability 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.[9]

For configuration examples of the Unified CVP application with VRU Type 8 or Type 2, refer to the latest version of the Cisco Unified CVP Configuration and Administration Guide, available at Cisco.com.

Network VRU Types and Unified CVP Call Flow Models

In Unified ICM, Network VRU is a configuration database entity. It is accessed using the ICM Configuration Manager's Network VRU Explorer tool. A Network VRU entry has two pieces of information:

Type: This is a number from 2 to 10 and corresponds to the types previously described.

Labels: This is a list of labels, which Unified ICM can use to transfer a call to the particular Network VRU that is being configured. These labels are only relevant for Network VRUs of Types 3, 7, and 10 (that is, those that use the Correlation ID mechanism to transfer calls). They are also required but never used in the case of Type 5. (Labels for Types 8 and 2 are defined in the ICM Configuration Manager's Translation Route Explorer tool and invoked via a Translation RouteToVRU node.) Each label is made up of two parts:

- A digit string, which becomes a DNIS that can be understood by the gatekeeper (when using H.323), by a SIP Proxy Server or static route table (when using SIP without a Proxy Server), SIP, or by gateway dial-peers.
- A routing client (also known as a switch leg peripheral). In other words, each peripheral device that can act as a switch leg must have its own label, even if the digit strings are the same in all cases.

As noted earlier, Unified ICM Release 7.1(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 Types 2, 3, 5, 7, or 8 Network VRUs, which were associated with the Customer Instance and the Switch and VRU leg peripherals. The VRU-Only call flow models still require Type 8. However, in one specific case Types 3 or 7 is still required.

Network VRU configuration entries 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 ICM Configuration Manager's PG Explorer tool
- In the customer Instance configuration in the ICM Configuration Manager's ICM Instance Explorer tool
- In every VRU Script configuration in the ICM Configuration Manager's Network VRU Script List tool

Depending on the call flow model, Unified ICM looks at either the peripheral or the customer instance to determine how to transfer a call to a VRU. Generally speaking, Unified ICM examines the Network VRU, which is associated with the switch leg peripheral when the call first arrives on a switch leg, and the Network VRU, which is associated with the VRU leg peripheral when the call is transferred to VRU using the Translation Route mechanism. It examines the Network VRU, which is associated with the Customer Instance or the default Network VRU from the System Information tool, when the call is transferred to the VRU using the Correlation ID mechanism.

Unified ICM also examines the Network VRU 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 does not execute the VRU Script.[10]

To examine how these VRU types interact with the previously defined Unified CVP Functional Deployment models, it is necessary to define the different variances of these models as such:

Model #1: Standalone Self-Service

Model #2: Call Director

Model #3a: Comprehensive Using ICM Micro-Applications

Model #3b: Comprehensive Using Unified CVP VXML Server

Model #4: VRU-Only

Model #4a: VRU-Only with NIC Controlled Routing

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

Model #1: Standalone Self-Service

As mentioned earlier in this chapter, this model does not interact with Unified ICM VRU scripts, so a Network VRU setting is not relevant. Even in the hybrid case in which the Standalone Self-Service model used Unified ICM for a label lookup, a VRU script is not invoked, only a simple Route Request to the VRU PG Routing Client. Therefore a Network VRU is not needed.

Model #2: Call Director

An earlier discussion pertaining to the Call Director model explained that Unified ICM (via Unified CVP) is responsible for call switching only. Because this model does not provide queuing or self-service, there is no VRU leg. Therefore, a Network VRU setting is not required.

Model #3a: Comprehensive Using Micro-Apps

In this model however, Unified CVP devices act as both the Switch and VRU leg, but interestingly enough, the call does not need to be transferred from the switch leg to the VRU leg before a call treatment can occur. Because this is the classic example of a Type 10 VRU, one should be associated to all the Unified CVP peripherals.

> **Note** Deployments using Unified ICM 7.0 and earlier configured these peripherals as a Type 2, creating interesting challenges about which CVP servers processed the Switch leg of a call and which were expected to handle the VRU leg.

In addition, all incoming dialed numbers should be associated to Customer Instance associated with a Type 10 Network VRU. All the VRU Scripts that will be executed by the incoming call must be associated with the same Type 10 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. This enables a VRU label to be generated and verify that the VoiceXML router can kick off and start the VRU leg of a call. By using this node in the routing script, an incremental step is provided testing the viability of the VRU components of the solution.[11]

Model #3b: Comprehensive Using Unified CVP VXML Server

From a call routing and Network VRU perspective, this model is identical to Model #3a previously described.

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 when it receives the call. This model actually has two submodels, which are described separately in the following sections.

Model #4a: VRU-Only with NIC Controlled Routing

This submodel assumes a fully functional NIC 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 requesting that the call be retransferred to another agent or back into queue or self-service, the NIC can retrieve the call from the agent and redeliver 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) cannot transfer a call to a particular destination directory number and carry an arbitrary Correlation ID along with it, which the destination device can pass back to Unified ICM to make the Correlation ID transfer mechanism properly function. For most NICs, therefore, the Translation Route mechanism must be used. There are a few exceptions to this rule, in which case the Correlation ID mechanism can be used.

The NICs that can transmit 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 can transmit the Correlation ID, the incoming dialed numbers must all be associated with a Customer Instance associated with a Type 7 Network VRU. The Type 7 Network VRU must contain labels 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 cannot transmit a Correlation ID (the usual and safe case), the incoming dialed numbers must all be associated with a Customer Instance 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.[12]

Model #4b: VRU-Only with NIC Controlled Prerouting

This submodel assumes a less capable NIC that can deliver the call only once, whether to a VRU or to an agent. When the call is delivered, the NIC cannot be instructed to retrieve the call and redeliver 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 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 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.[13]

Note Two different VRU transfer nodes are required. The first one transfers the call away from the NIC with a handoff. 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.

Summary

Unified CVP has a significant amount of flexibility in how it is deployed as discovered in this chapter. The different functional deployment models discussed provide a simple set of architectural starting points for engineers to understand these deployments and to discuss their strengths and differences with their customers. This chapter also discussed interactions with Unified ICM to provide some basic integration concepts about the switch and VRU leg of a call and how Unified ICM deals with each.

In addition, this chapter provided a detailed overview of the different call flows supported by each function deployment model solidifying this knowledge for use in future chapters pertaining to high-availability designs and troubleshooting the solution. As mentioned earlier, it is critical that engineers understand the solutions components, different deployment models, and their respective call flows to have a solid base of solution architectural knowledge to build upon with more advanced features found in later chapters.

The next chapter explores how CVP handles different geographical deployments such as centralized and distributed branches. It also illustrates how a distributed branch or edge design can be accomplished using different techniques supported by Unified CVP with SIP and H.323. It also acknowledges the importance of understanding the geographical deployments supported by CVP and how to build high availability into each of these components depending on their geographical placement.

References

1. Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, 2010: 2–3.

2. Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, 2010: 2–4.

3. Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, 2010: 2–6.

4. Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, 2010: 2–11.

5. Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, 2010: 5–3.

6. Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, 2010: 5–3.

7. Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, 2010: 5–4.

8. Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, 2010: 5–5.

9. Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, 2010: 5–6.

10. Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, 2010: 5–7.

11. Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, 2010: 5–8.

12. Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, 2010: 5–8, 9.

13. Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, 2010: 5–9.

Recommended Reading and Resources

Cisco Documentation, *Cisco Unified Customer Voice Portal (CVP) 8.x Solution Reference Network Design (SRND)*, http://www.cisco.com/en/US/docs/voice_ip_comm/cust_contact/contact_center/customer_voice_portal/srnd/8x/cvp8xsrnd.pdf.

Cisco Documentation, *Cisco Unified Contact Center Enterprise Solution Reference Network Design (SRND)*, http://www.cisco.com/en/US/products/sw/custcosw/ps1844/products_implementation_design_guides_list.html.

Cisco Documentation, *Cisco Unified Customer Voice Portal Configuration Guides*, http://www.cisco.com/en/US/partner/products/sw/custcosw/ps1006/products_installation_and_configuration_guides_list.html.

Cisco Documentation, *User Guide for Cisco Unified CVP VXML Server and Cisco Unified Call Studio, Release 8.0(1)*, http://www.cisco.com/en/US/docs/voice_ip_comm/cust_contact/contact_center/customer_voice_portal/cvp8_0/user/guide/cvp_801_vxml.pdf.

Cisco Documentation, *Hardware and System Software Specification for Cisco Unified Customer Voice Portal (Unified CVP)*, Release 8.0(1), http://www.cisco.com/en/US/docs/voice_ip_comm/cust_contact/contact_center/customer_voice_portal/cvp8_0/reference/guide/cvp_801_bom.pdf.

Cisco Documentation, *Cisco Customer Contact Solutions Ordering Guide*, http://www.cisco.com/en/US/partner/prod/collateral/voicesw/custcosw/ps5693/ps1844/CCBU_ordering_guide.pdf.

Cisco Documentation, *Cisco ACE 4700 Series Appliance Server Load-Balancing Configuration Guide*, http://www.cisco.com/en/US/docs/app_ntwk_services/data_center_app_services/ace_appliances/vA3_1_0/configuration/slb/guide/slbgd.pdf.

Cisco Documentation, *Cisco CSS 11500 Series Content Services Switches Configuration Guides*, http://www.cisco.com/en/US/partner/products/hw/contnetw/ps792/products_installation_and_configuration_guides_list.html.

Index

A

D

E

F

G

H

I

O-P

Q

R

S

T

U

V

W

X-Y-Z