Oracle® Web Conferencing

Administrator's Guide

Release 2 (2.0.4.3)

Part No. B10877-03

August 2004

Use this manual to deploy, configure, and administer Oracle Web Conferencing.

ORACLE®

Oracle Web Conferencing Administrator's Guide, Release 2 (2.0.4.3)

Part No. B10877-03

Primary Author: Jeffrey Doering, Dinesh Ajmera, PV Shivkumar, Rami Musa, Barbara Heninger

Contributing Author: Anindo Roy, Mehul Snghavi, Kiran Choudary

Contributor: Ramu Sunkara, Vimal Chopra, Michael Tselner, Ravindra Dani

Contents

3 Planning for Deployment

4 Post-Installation

5 Configuration

6 Sample Deployments

7 Monitoring

8 Reports

9 Web Conferencing Sites

10 imtctl Command Line Utility

List of Figures

Send Us Your Comments

Oracle Web Conferencing Administrator's Guide, Release 2 (2.0.4.3)

Part No. B10877-02

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: realtimecollab_us@oracle.com
- FAX: 650-506-7415. Attn: Oracle Web Conferencing Documentation
- Postal service:

  Oracle Corporation
  Oracle Web Conferencing Documentation
  500 Oracle Parkway, Mailstop 2op745
  Redwood Shores, California, 94065
  United States

If you would like a reply, please give your name, address, telephone number, and electronic mail address (optional).

If you have problems with the software, please contact your local Oracle Support Services.

Preface

This preface describes the intended audience, content, documentation conventions and other features of the *Oracle Web Conferencing Administrator's Guide.*

Intended Audience

This manual is intended for administrators who must install, configure, and monitor Oracle Web Conferencing. The Administrator's Guide is not intended for end users.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at

`http://www.oracle.com/accessibility/`

Accessibility of Code Examples in Documentation

JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Structure

This document consists of the following chapters:

Chapter 1 Introduction to Oracle Web Conferencing

This chapter provides an overview of the Oracle Web Conferencing product.

Chapter 2 Understanding Oracle Web Conferencing

This chapter explains Oracle Web Conferencing concepts, including terms and definitions, architecture, processes, conference runtime flow, and connection details.

Chapter 3 Planning for Deployment

This chapter discusses factors to consider when deploying Oracle Web Conferencing. The issues range from sizing guidelines to firewall issues to topology considerations for an enterprise that is geographically distributed.

Chapter 4 Post-Installation

This chapter discusses mandatory post-installation steps required to make Oracle Web Conferencing completely functional in the selected deployment configuration.

Chapter 5 Configuration

This chapter explains how to configure the Web Conferencing system, including ports and network connectivity, user privileges, SSL security, clusters, e-mail invitations to conferences, Web Conferencing application pages, conference runtime features, Voice Conversion servers, logging, and system default time zones.

Chapter 6 Sample Deployments

This chapter discusses some sample deployment scenarios, ranging from a basic deployment to a sophisticated one spanning multiple geographical regions. It also explains how to configure these deployments.

Chapter 7 Monitoring

This chapter describes how Oracle Web Conferencing components can be monitored to provide quality of service for conferences and continuous availability for conference service.

Chapter 8 Reports

This chapter explains the Usage and Feedback reports.

Chapter 9 Web Conferencing Sites

This chapter explains Oracle Web Conferencing sites.

Chapter 10 imtctl Command Line Utility

This chapter explains the imtctl utility, which provides a command-line interface for administering and configuring the Real-Time Collaboration system.

Chapter 11 Troubleshooting

This chapter provides a URL where you can find the most up-to-date troubleshooting information.

Glossary

This manual contains a glossary. Terms that are explained in the glossary appear in bold in the text.

Appendix A Time Zones

This appendix contains a list of supported time zones.

Related Documents

For more information, see the following manuals in the Oracle Collaboration Suite Release 9.0.4.2.0 documentation set:

- *Oracle Collaboration Suite Installation and Configuration Guide for Solaris*
- *Oracle Collaboration Suite Release Notes*
- *Oracle Web Conferencing Sizing Guide*
- *Oracle Collaboration Suite Using Web Conferencing*

Conventions

The following conventions are also used in this manual:

Convention	Meaning
. . .	Vertical ellipsis points in an example mean that information not directly related to the example has been omitted.
. . .	Horizontal ellipsis points in statements or commands mean that parts of the statement or command not directly related to the example have been omitted
boldface text	Boldface type in text indicates a term defined in the text, the glossary, or in both locations.
< >	Angle brackets enclose user-supplied names.
[ ]	Brackets enclose optional clauses from which you can choose one or none.

1

Introduction to Oracle Web Conferencing

The Oracle Real-Time Collaboration system is a state-of-the-art, distributed system that offers real time collaboration services, including Web Conferencing. While this guide focuses on the administration and management of the Web Conferencing service, most of the components and administration tools that are involved in the providing this service are generic and will be used for other services in the future.

This chapter provides an overview of the Oracle Web Conferencing product, including its end-user collaboration features and administration features.

What is Oracle Web Conferencing?

Oracle Web Conferencing brings real-time online collaboration to any enterprise, letting customers, employees, teams, and partners meet online within the context provided by the content, commerce, and comprehensive business flows of e-business.

Oracle Web Conferencing consists of client and server applications that let you create and participate in online conferences. From a user's perspective, there are two main parts to the Web Conferencing system:

- **The Web Conferencing Application** A series of pages displayed in the web browser that let users schedule or join conferences, replay past conferences, or select materials to be shared during future conferences. System administrators can also use the pages to monitor the status of the Web Conferencing installation and review statistics about past conferences, such as connection time or number of concurrent users.

Figure 1–1 Web Conferencing Application: System Administrator's View

Note: In the preceding figure, the **Monitor** and **Reports** tabs are only available to users assigned the business monitor or business administrator role. The **Sites** and **System** tabs are only available to users assigned the business administrator role. See "User Management" on page 2-10 for more details about user privileges.

- **The Web Conferencing Console** Users interact in the console during a conference, sharing their desktops, cobrowsing web pages, using an online "whiteboard," chatting, and polling conference attendees.

Figure 1–2 Web Conferencing Console Banner

The following sections describe the features of the Web Conferencing Application and the Web Conferencing Console in more detail.

Web Conferencing Application Features

The Oracle Web Conferencing Application (Figure 1–1) lets users start and manage their conferences. Users may:

- *Schedule a conference*: Plan a conference in advance, send e-mail invitations, and designate materials to be reviewed before the conference.
- *Create an instant conference*: Create a conference and host it right away.

- *Join a conference*: Easily join a conference through an e-mail invitation (if the host has enabled this option), through the lists of conferences, or through the Join Meeting window.
- *Manage materials*: Store conference materials such as documents, bookmarks, messages, and polls in your own repository. Users can access their personal materials repository during conferences.
- *Record a conference*: Record a conference so that it can be played back anytime.
- *Play back a conference*: Play back archived conferences any time.
- *Publish archives*: Publish the recorded conference and information about the conference, such as the public chat transcript, list of conference attendees, and conference duration.

System administrators also use the Web Conferencing application to monitor system performance, create custom sites, and display reports about the system performance. See "Web Conferencing Management Features" on page 1-6 for an overview of administration and management features.

Web Conferencing Console Features

The Web Conferencing console (Figure 1–2) is where online conferences take place. Each conference participant runs the console on his or her desktop. The console features include:

Cobrowsing

- Synchronous browsing and scrolling of HTML pages with a small or a large number of users.
- Navigation to Web pages by typing a Web address during a conference or by selecting from user-defined bookmarks.
- Filling out web forms, including support for real-time individual character display.

Document Presentation

- Selection and display of documents from a user's personal materials repository. Presentation of the following formats is supported: Microsoft Word, Excel, PowerPoint, HTML, text and image (.gif and .jpg).

Whiteboarding

- Mark-up and drawing tools for the visual review and annotation of shared content, or as an aid in making presentations.

Desktop Sharing

- Sharing of applications from the presenter’s Microsoft Windows-based computer with other attendees in real time. Options include sharing any region or multiple regions, any application, or the entire desktop.

Voice Streaming

- Live, listen-only streaming of voice from any telephone or teleconference system through the Oracle Web Conferencing Console.

Polling

- Ability to create instant polls or select user-predefined polls. Responses are displayed in real time.
- Publishing of poll results to all attendees.

Chat

- Live chat with all attendees, a group of attendees, or with just the conference host.
- Ability for registered users to predefine messages for use during live chat.

Shared Control

- Shared control of the conference with attendees. (The conference host always has ultimate control.)

Recording and Playback

- Recording of voice synchronized with any on-screen data collaboration for on-demand playback of the conference.

Oracle Web Conferencing Components

Oracle Web Conferencing is an option of Oracle Real-Time Collaboration. A Web Conferencing system is made up of the following:

Oracle Web Conferencing Console

Conferences take place in the Web Conferencing Console (shown in Figure 1–2. When a user joins a conference, the Web Conferencing Console is downloaded if it is not already present on the user's computer or if the installed version is not current. Users access the Web Conferencing Console through Internet Explorer 5.5 or higher or Netscape 4.75 or higher. (Not all features of Web Conferencing are available when using Netscape; users can only attend conferences, not schedule them, and some console features such as Chat are disabled.)

Oracle Real-Time Collaboration Core Components

The Real-Time Collaboration core components include the Web Conferencing Server, the Web Conferencing Application (shown inFigure 1–1, and the Oracle Real-Time Collaboration process monitor and the Real-Time Collaboration multiplexer. These components work together to provide the core real-time collaboration functionality of Oracle Web Conferencing.

As a prerequisite, the Oracle Real-Time Collaboration Core Components need an Oracle9*i*AS mid-tier that is configured to work with Oracle9iAS Single Sign-On and Oracle Internet Directory. See "Web Conferencing Prerequisites" for details.

Document Conversion Server

A group of Oracle Web Conferencing processes that converts MS Office documents into HTML for viewing in Document Presentation mode. The Document Conversion Server component must be installed on a computer with Microsoft Windows and Microsoft Office.

> **See Also:** *Oracle Collaboration Suite Installation and Configuration Guide for Solaris* for details about prerequisites

Voice Conversion Server

A group of Oracle Web Conferencing processes that dials into a voice conferencing system, converts the analog voice to digital format, and streams it. The Voice Conversion Server component must be installed on a computer with Microsoft Windows and requires specialized telephony hardware and software.

> **See Also:** Oracle Collaboration Suite Installation and Configuration Guide for Solaris for details about prerequisites

Real-Time Collaboration Repository

The set of Oracle Real-Time Collaboration database schemas residing in an Oracle9i Database .

- **rtc**: Contains all the database tables, views, indexes, triggers, etc. This account is not used by Oracle Real-Time Collaboration components.
- **rtc_app**: Contains the appropriate synonyms pointing to base Oracle Real-Time Collaboration tables and views that belong in the rtc account. This account has appropriate privileges to modify and select data from the tables. This account is used for connection by Oracle Web Conferencing components.

The schemas contain:

- **System map**: Configuration information for all the instances and components in the system.
- **User roles**: Roles assigned to users setting system privileges. The user authentication information itself is stored in Oracle Internet Directory.
- **Conference information**: Scheduled conferences, conference archives, user documents loaded for use during a conference.
- Data for internal use by Oracle Real-Time Collaboration, such as application menu definitions.

Web Conferencing Prerequisites

The following prerequisites are necessary for any deployment of Oracle Web Conferencing.

Oracle9*i*AS

Oracle9i Application Server is an integrated J2EE application server that provides the Oracle HTTP Server, Oracle9iAS Containers for J2EE, and other Oracle Web Conferencing prerequisites.

Oracle9*i*AS Infrastructure

This includes Oracle Internet Directory and Oracle9iAS Single Sign-On. Among its services, Oracle9*i*AS Infrastructure provides user provisioning and authentication services.

Oracle9*i* Database

The Oracle9*i* Database, release 2, is a prerequisite for Oracle Web Conferencing. The Oracle Real-Time Collaboration Repository for Oracle Web Conferencing resides in this database.

Web Conferencing Management Features

Oracle Web Conferencing management consists of system management, business management, and user management.

System Management Features and Tools

Web Conferencing offers the following features for managing the overall system:

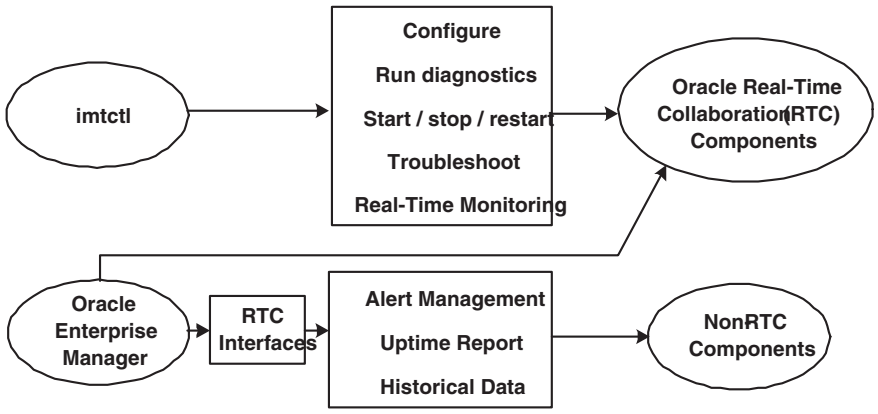
Support for management of different topologies—Lets you manage Web Conferencing in a geographically-distributed enterprise and provides support for various deployments, taking into account all enterprise considerations like firewall issues and load balancers (LBR).

High availability—Provides process monitoring and automatic restart of processes when component failure is detected.

Oracle Enterprise Manager—Integrated with Oracle Enterprise Manager.

Figure 1–3, "Real-Time Collaboration System Management Tasks" illustrates Real-Time Collaboration system management:

Figure 1–3 Real-Time Collaboration System Management Tasks



The `imtctl` utility provides a command-line interface for administering and configuring the Real-Time Collaboration system. The utility supports a variety of commands:

- Configuring system attributes, such as features available in conferences.
- Running diagnostics of the system.
- Starting, stopping, or restarting Real-Time Collaboration instances.
- Troubleshooting system issues.
- Monitoring the system.

Oracle Enterprise Manager is used to manage the external components on which Oracle Real-Time Collaboration depends, such as the Oracle9*i*AS mid-tier, Oracle9*i*AS Infrastructure, Oracle9*i* Database. Real-Time Collaboration interfaces are used for monitoring Web Conferencing and generating alerts.

The Oracle Enterprise Manager standalone console also provides basic instance management operations, like starting and shutting down an instance and viewing the

state of the instance. Refer to the Oracle Enterprise Manager *Administrator's Guide* for more information.

Real-Time Collaboration Interfaces are available to plug into any external management framework. With the interfaces, you can run service availability tests and get real-time monitoring data that can be collected periodically and used for historical analysis to tune the system. See Chapter 7, "Monitoring" for more details about these interfaces.

Business Management

Oracle Real-Time Collaboration business management provides a rich set of features that business managers and support staff can use to monitor usage and manage business-related activities.

Conference Monitoring—You can monitor conferences that are currently running on the system. This is useful both in support situations and business administration. See Chapter 7, "Monitoring" for more details.

Reporting—Oracle Real-Time Collaboration provides comprehensive usage and feedback reports. See Chapter 8, "Reports" for more details.

Site Management—You can customize a single Oracle Web Conferencing deployment in an enterprise for different lines of business or site usage. Each line of business can have its own look-and-feel and customized integration with the Real-Time Collaboration system. See Chapter 9, "Web Conferencing Sites" for details.

Monitoring and reporting features are available in the Web Conferencing Application interface to users with appropriate Web Conferencing administration privileges. Administrators can also use the imtctl command-line interface to perform various business management tasks. Using the ReportManager infrastructure, you can generate reports that can be automatically e-mailed to recipients.

User Management

The Real-Time Collaboration system uses Oracle Internet Directory to manage user data. The Oracle Internet Directory host used by Real-Time Collaboration is specified at installation. All users of this Oracle Internet Directory are automatically provisioned to use Real-Time Collaboration with the enduser role.

By default, users are managed using the oiddas interface of Oracle Internet Directory. This is typically available at `http://<ldaphostname>:7777/oiddas`. A user with the Oracle Internet Directory administrator account, typically *orcladmin*, can create, update, and delete users.

> **See Also:** *Oracle Collaboration Suite Installation and Configuration Guide for Solaris* for more information

Administration Road Map

To successfully deploy Oracle Web Conferencing and manage the system, Oracle Corporation recommends following the steps.

Step 1 Understand Oracle Web Conferencing.

Understand the basic concepts of the Oracle Web Conferencing product, including its architecture, components, end-user connections, load balancing, and network and port considerations for deployment. See this chapter and Chapter 2, "Understanding Oracle Web Conferencing" to familiarize yourself with Web Conferencing architecture and components.

Step 2 Plan for deployment.

Follow the sizing guidelines in the *Oracle Web Conferencing Sizing Guide*to estimate the hardware requirements for your deployment. Take into account the existing deployment of other Web-based applications, including firewall considerations and load balancers by reading

Step 3 Install Oracle Web Conferencing.

Install the Oracle Real-Time Collaboration system, following the instructions in the *Oracle Collaboration Suite Installation and Configuration Guide for Solaris*.

Note: The Installation Guide contains information for installing all Oracle Collaboration Suite tools. If you are only installing Web Conferencing, read the first sections about general installation, and the Web Conferencing installation section.

Then follow the post-installation steps in Chapter 4, "Post-Installation". When you are finished, run the post-installation verification tests described in the chapter. Out of the box, some tests might fail. The most common reason is incomplete configuration. Fix the configuration for each of the failures.

Step 4 Monitor Oracle Web Conferencing.

Monitor the health of the Real-Time Collaboration system by running service availability tests periodically. Have the results of the tests tied to an alert management system. See Chapter 7, "Monitoring" for details.

Step 5 Troubleshoot Oracle Web Conferencing.

Troubleshoot the system as required to address user complaints and failures detected by diagnostic tests run manually or through alerts. Use the information available from the **Monitor**, **Reports**, and **System** tabs to assist you.

Step 6 Tune the Web Conferencing system.

Tune the Web Conferencing (the Real-Time Collaboration) system, as required. You might need to change the existing configuration or add more machines.

2

Understanding Oracle Web Conferencing

This chapter explains Oracle Web Conferencing concepts and architecture. This chapter describes:

- Real-Time Collaboration terms and concepts
- Real-Time Collaboration processes of Oracle Web Conferencing, including the function of each and their interactions.
- The detailed runtime flow of various kinds of clients connected to the Web Conferencing Server for a conference.
- Ports required for Web Conferencing and and an overview of network connectivity issues

Concepts and Terminology

The following sections describe in detail some of the terms used in Oracle Real-Time Collaboration and introduce you to some basic concepts about the system.

Real-Time Collaboration Instance

A Real-Time Collaboration instance is a group of Real-Time Collaboration components installed on the same machine.

Three basic components—the Core Components, Document Conversion Server, and the Voice Conversion Server—are created during installation. If all are installed on the same machine, there is only one instance only. The Real-Time Collaboration Repository is either present in the information store database when Oracle Collaboration Suite information store is installed, or it is created during installation of first Core Components installation.

> **See Also:** *Oracle Collaboration Suite Installation and Configuration Guide for Solaris* for details.

Real-Time Collaboration Component

An Oracle Real-Time Collaboration component is a set of processes within an instance that perform an identical function. Each component has a component type and component name. (Component types and names can be used when configuring the system using the imtctl command, as described in Chapter 10, "imtctl Command Line Utility".)

The following table lists the Real-Time Collaboration component types and names:

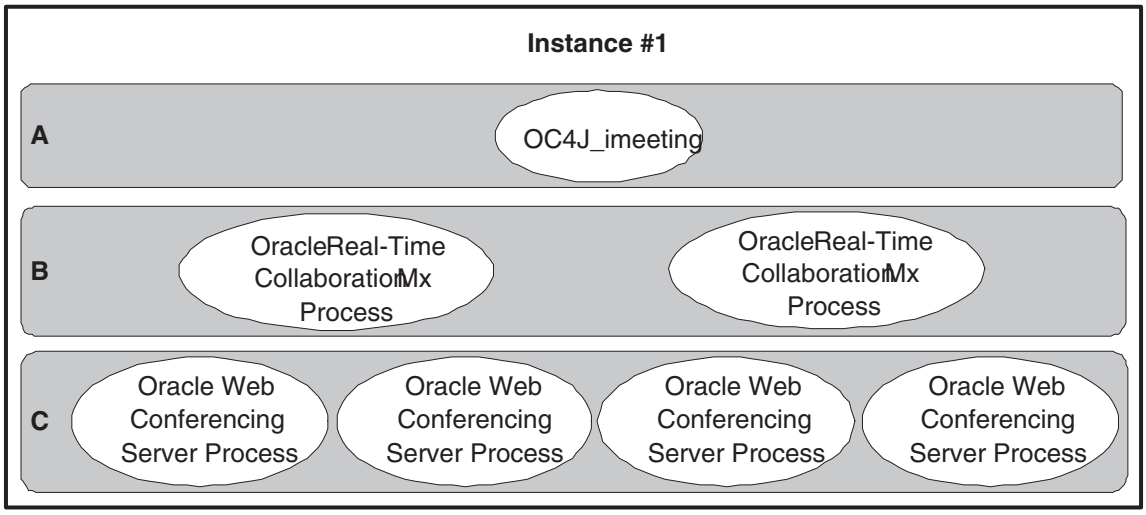
Table 2–1 Real-Time Collaboration Components

Full Component Name	Component Type	Component Name	Number of Processes
Oracle Web Conferencing Server	clbsvr	imt-collab	1-n
Web Conferencing J2EE Application: OC4J_imeeting	oc4j	oc4j_imeeting	1-n
Multiplexer	mxcomm	imt-mx	1-n
Voice Conversion Server	voiceconv	imt-voiceconv	1
Document Conversion Server	docconv	imt-docconv	1
Real-Time Collaboration Process Monitor	imt-pm	imt-pm	1

You can use the `listComponents` imtctl command to see a list of components in an instance. See Chapter 10, "imtctl Command Line Utility" for details.

Figure 2–1, "Components and Processes of a Real-Time Collaboration Instance" shows an instance with three components and each of their processes.

Figure 2–1 Components and Processes of a Real-Time Collaboration Instance



Component A: OC4J_imeeting (one process)

Component B: Real-Time Collaboration multiplexer (two processes)

Component C: Web Conferencing Server (four processes)

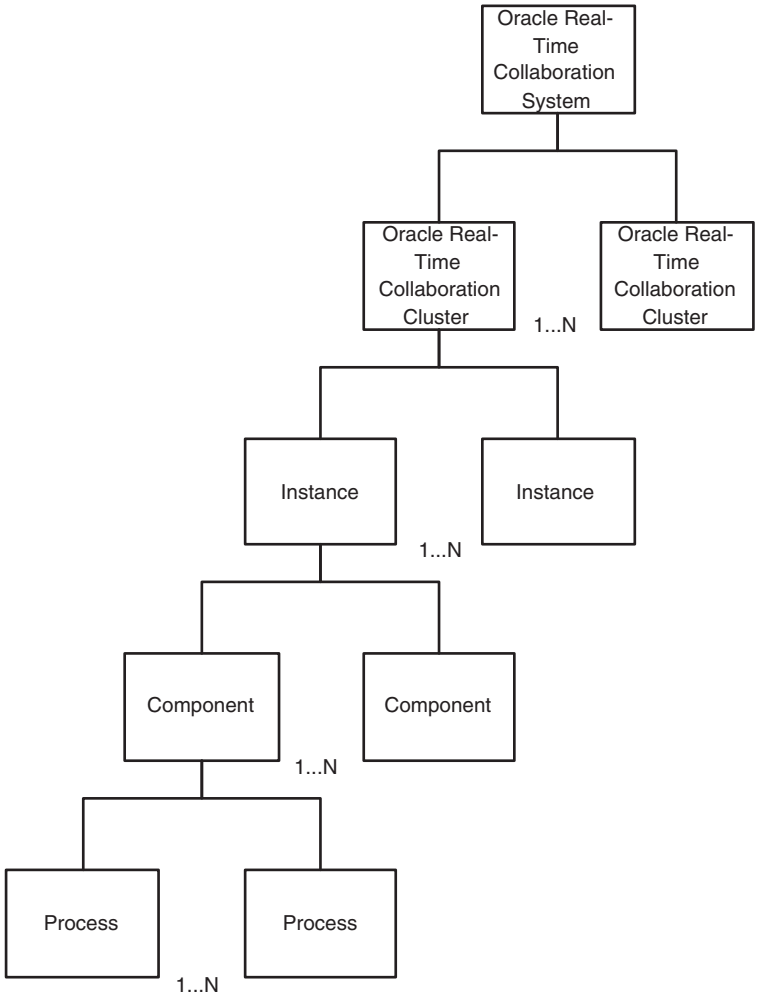
Real-Time Collaboration Cluster

A group of Real-Time Collaboration instances that have the same `InstanceLocation` property and that use the same Real-Time Collaboration Repository. Clusters are discussed in more detail in Chapter 3, "Planning for Deployment".

Real-Time Collaboration System

The set of *all* instances that share the same Oracle Real-Time Collaboration Repository. Because instances can be part of clusters, the Oracle Real-Time Collaboration system can be thought of as a set of all clusters.

Figure 2–2 Real-Time Collaboration System Hierarchy



Functions of Real-Time Collaboration Components

Figure 2–3 Real-Time Collaboration Architecture

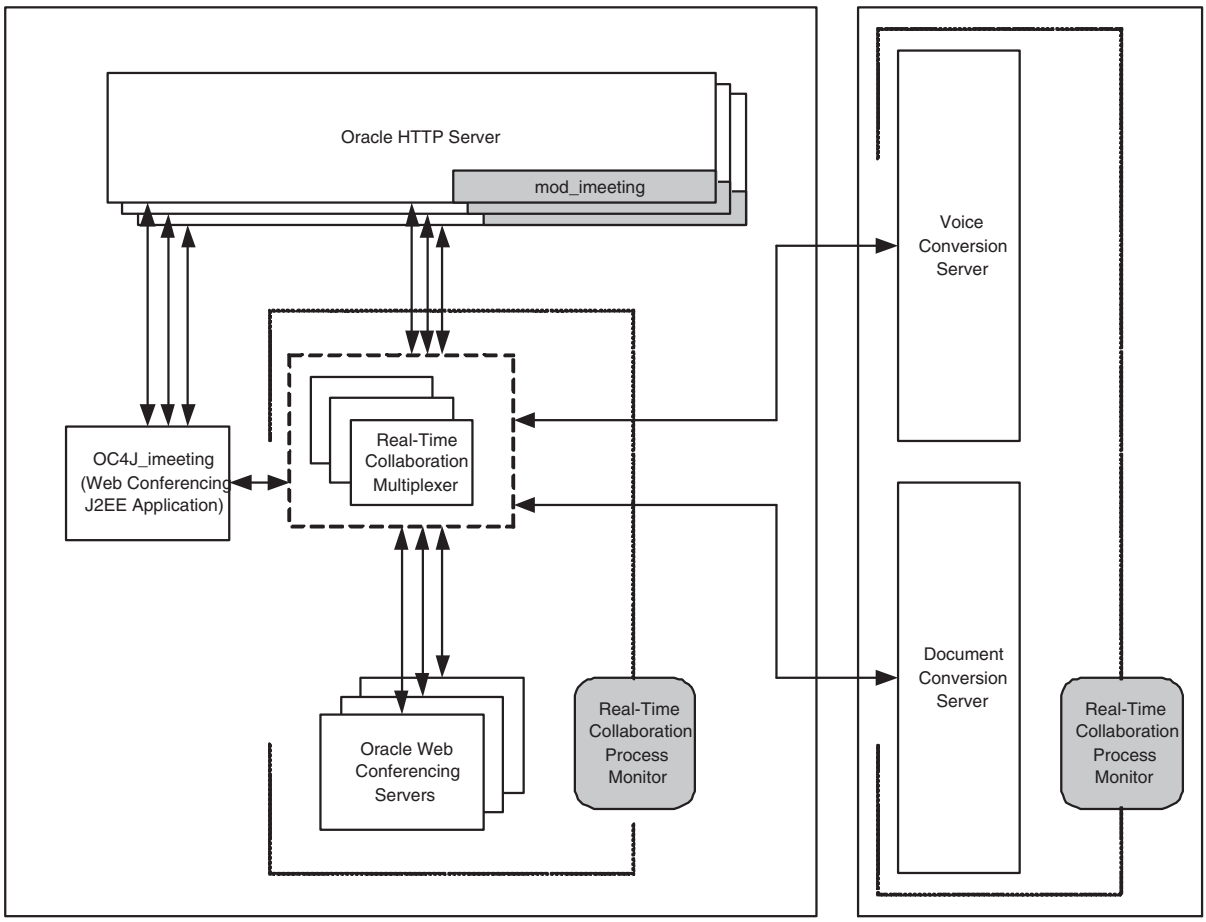


Figure 2–3, "Real-Time Collaboration Architecture" shows how the Real-Time Collaboration components interact with each other. The following sections describe each component's role in the Web Conferencing system in more detail.

Oracle Web Conferencing Server

A Web Conferencing Server component does the following:

- Manages all the conference attendees' states and their permissions within the conference
- Intelligently distributes real-time data for all the collaboration modes that are active during the conference
- Provides services for the recording and archiving of the conference

Real-Time Collaboration Multiplexer (mx)

The multiplexer component does the following:

- Accepts inbound connections from clients, Web Conferencing Servers, and other Real-Time Collaboration processes.

- Routes data traffic between all clients and all Web Conferencing Servers on a machine.
- Acts as a communication hub for all components.

OC4J_imeeting

OC4J_imeeting is the Oracle Web Conferencing J2EE Application running in the Oracle9iAS Containers for J2EE. It does the following:

- Provides the Web-based user interface to Oracle Web Conferencing for end-users.
- Provides integration with external applications like Oracle Calendar.
- Interfaces with Oracle Internet Directory for user management.

Real-Time Collaboration Process Monitor (imt-pm)

The process monitor watches all the other Real-Time Collaboration processes (except for OC4J_imeeting processes), and provides high availability by restarting processes if they are down. This process itself is monitored by the Oracle Process Management and Notification system.

Voice Conversion Server

The Voice Conversion server dials in to a voice conferencing system, captures the analog voice stream, digitizes it, and streams it to a Web Conferencing Server process through the mx.

Document Conversion Server

The Document Conversion server converts Microsoft Office documents to HTML for document presentation. The Web Conferencing Application provides the front end for a user to convert a document.

HTTPD + mod_imeeting

The mod_imeeting plug-in interacts with the Oracle HTTP Server. It makes it possible for Oracle Web Conferencing to use Oracle HTTP Server as the only listening point to listen for end-user connection requests. Connections are handed off to the mx using socket hand-off, after the connection has been established using Oracle HTTP Server as the listener. See "Client Connection Details" on page 2-7 for more information about how the HTTP Server and plug-in interact.

Process Interaction

The following sections describe the Real-Time Collaboration processes within an instance, and across instances. They also describe the connections made by the processes.

Process Interaction Within an Instance

Within an instance, each Web Conferencing Server process is always connected to each multiplexer (mx) process.

Each OC4J process can connect to any mx in the cluster to which the instance belongs. OC4J initially connects to one of the mx processes when required, and from then on caches the connection.

Through the Oracle Web Conferencing Console, each conference participant connects to one of the mx processes, either directly or through a socket hand-off as described in

"Client Connection Details" on page 2-7. The mod_imeeting connects to all the mx processes in the same instance.

Process Interaction Across Instances

Across instances, the Voice Conversion server and Document Conversion server processes in an instance connect to all the multiplexer processes in another instance that they have been configured to service.

Connections

Real-Time Collaboration components make both virtual channel connections and connections to the Real-Time Collaboration Repository (the database).

Virtual Channels

All processes connect to the multiplexer, which acts as a communication hub, and so the following essential virtual channels are created:

- The Web Conferencing Console connects to the Web Conferencing Server process through the multiplexer during a conference.
- The Oracle Web Conferencing OC4J_imeeting processes connect to the Document Conversion Server process through the multiplexer used for document conversions.
- Each Web Conferencing Server process connects to the Voice Conversion Server process through the multiplexer for voice streaming during a conference.

Database Connections

- Each Web Conferencing Server process maintains a pool of connections to the Real-Time Collaboration Repository.
- Each OC4J_imeeting process maintains a pool of connections to the Real-Time Collaboration Repository.
- Each Document Conversion Server maintains a connection to Real-Time Collaboration Repository.
- Each Real-Time Collaboration Process Monitor maintains a connection to Real-Time Collaboration Repository.

Runtime Flow

The following sections describe the flow of events as users join a conference, and provide more details about how users connect to the Web Conferencing system.

Join Conference Flow

When an attendee joins a conference through the Oracle Web Conferencing Application, an e-mail invitation, or through an application that is integrated with Oracle Web Conferencing, the following events take place:

1. The application looks up the conference record from the database and retrieves the list of Web Conferencing Servers that can host the conference. The application authorizes the request based on the person joining and the attributes of the conference the person intends to join.

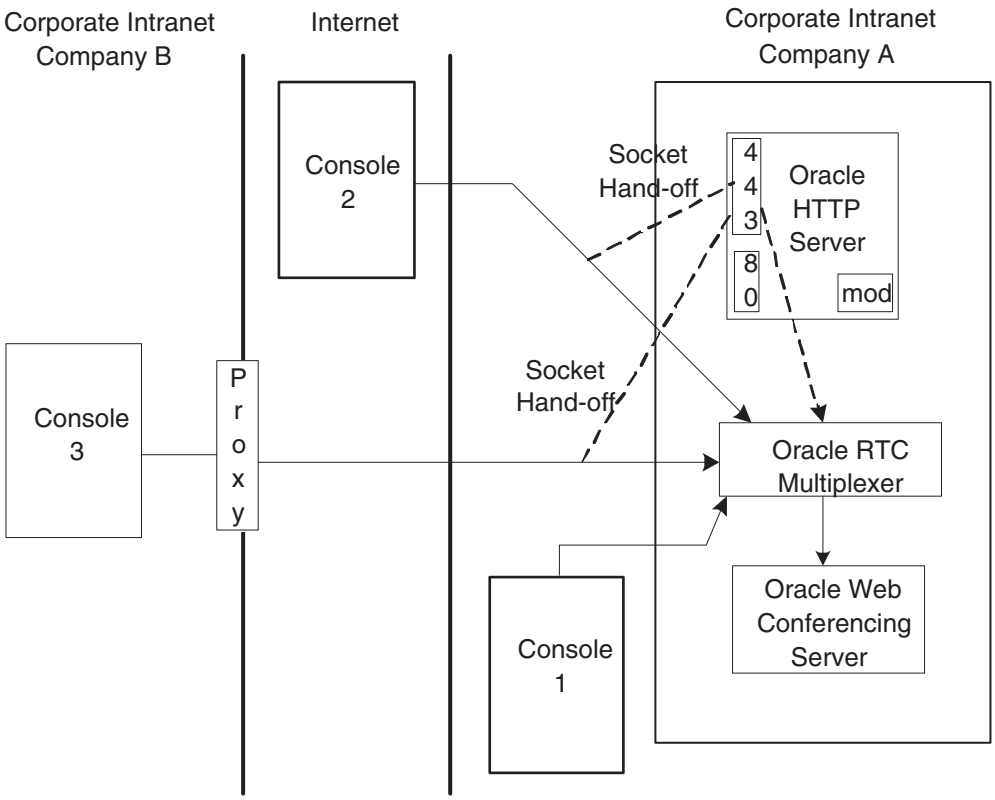
2. The server load balancer functionality in the Oracle Web Conferencing Application (OC4J_imeeting) chooses a Web Conferencing Server process, initiates the session there, and records the association between the conference ID and the Web Conferencing Server process.
3. The list of client parameters for the Web Conferencing Console to connect is generated:
   - The multiplexer load balancer (running in OC4J) functions in the Oracle Web Conferencing Application (OC4J_imeeting). It chooses one mx, which can be used to connect to that server process.
   - HTTPS connection information is taken from the mx description (that identifies which Oracle HTTP Server/mod can be used to redirect the connection).
   - An encrypted client authentication token is generated.
   - URLs to send user feedback information and retrieve Java/JSP components used in the console are generated.
4. The Oracle Web Conferencing Application response causes a pop-up window on the user's Web browser to open, which contains a Web Conferencing Console Installer (an ActiveX control) with the client parameters.
5. If the user does not have the Web Conferencing Console Installer yet or has an earlier version, the most recent version is downloaded and installed by Internet Explorer automatically (with a permission alert).
6. The Web Conferencing Console Installer performs compatibility checks. If compatibility checks are successful, it checks the version of the Web Conferencing Console available on the client system.
7. If the Web Conferencing Console is not installed on the client machine, or if the Web Conferencing Console version does not match the current one, the new console package is downloaded and installed.
8. The Web Conferencing Console Installer starts the Web Conferencing Console (as a separate process) with all parameters.
9. The Web Conferencing Console tries to establish connection to mx using the algorithm described in detail in "Client Connection Details". Then:
   - If all attempts fail, the client receives an error message.
   - If a connection is established, then the Web Conferencing Console creates a virtual channel through mx to the conference session and continues with the next steps.
10. The Web Conferencing Console sends an authorization token identifying the client to the server.
11. The Web Conferencing Server sends all conference state information (list of attendees, shared content, chat transcripts, and so on) to the Web Conferencing Console to initialize the Console.
12. The Web Conferencing Console opens and the conference begins.

Client Connection Details

This section describes in more detail how attendees connect to Web Conferencing. All conference attendees fall into one of three categories:

- Attendees connect directly to Oracle Web Conferencing without traversing any firewall. Example: all attendees are in the corporate intranet.
- Attendees connect to Oracle Web Conferencing from the Internet crossing a company firewall. Example: Web Conferencing is deployed in a company demilitarized zone (DMZ), and attendees from the Internet connect to the conference.
- Attendees are connecting through the Internet from another company's corporate intranet through their proxy.

Figure 2–4 Client Connections to Web Conferencing



The Web Conferencing Console attempts to connect to the multiplexer as follows:

1. **Direct TCP/IP**: This method is typically successful for clients within a corporate intranet. It is also successful for clients coming from the Internet, if the multiplexer port is open to the Internet.
2. **HTTPS direct** (through Oracle HTTP Server/mod_imeeting): If direct TCP/IP fails, the Web Conferencing Console tries to connect through HTTPS. This connection is typically successful for clients in the open Internet or across transparent proxies. Once a connection is established by Oracle HTTP Server, it is handed off to the multiplexer by mod_imeeting using socket hand-off. The multiplexer and the Web Conferencing Console then communicate directly with each other.
3. **HTTPS tunnel** (through Oracle HTTP Server/mod_imeeting): If both direct TCP/IP and HTTPS direct fail, going through the HTTPS tunnel is the only connection method for clients in a different intranet coming through their own internal proxy. The Web Conferencing Console tries to retrieve proxy information

from the browser settings on the client machine and establish a connection to the Oracle HTTP Server using the proxy. Once established, the connection is handed off to the multiplexer by mod_imeeting. In this case, the Web Conferencing Console and multiplexer communicate over the HTTPS tunnel through the remote proxy.

The connection information required to connect using the methods described in this section is provided to a Web Conferencing Console (transparent to a user) when a user tries to join a conference.

Ports and Network Connectivity

The following table contains port and network connectivity information for Oracle Web Conferencing.

Table 2–2 Ports and Network Connectivity

Component	Protocol	Port	IP	No. of Ports	Accessibility (Mandatory)	Accessibility (Recommended)
Oracle HTTP Server / mod_imeeting	HTTP	80	Primary	1	All clients	
Oracle HTTP Server / mod_imeeting	HTTPS	443	Primary	1	All clients	
Oracle HTTP Server / mod_imeeting	HTTPS tunnel	443	Secondary[1]	1	All clients	
mx (multiplexer)	mx[2]	2400-49151[3]	Primary	n	Voice Conversion Server and Document Conversion Server must be able to access the multiplexers on the instances they serve.	Intranet clients could connect using direct TCP/IP. If Real-Time Collaboration Core Components are deployed in a DMZ, accessing these port(s) from the intranet is not an issue.
mx (on NT)	redirect[4]	2400-49151	Primary	n	Local host	
voiced	HTTP	2400-49151	Primary	1	All machines with the Real-Time Collaboration Core Components this Voice Conversion Server supports.	For remote status
imt-pm	HTTP	2400-49151	Primary	1	Local host from all Real-Time Collaboration instances.	

[1] Second IP address required only if Oracle9iAS Web Cache is present on the machine with the Real-Time Collaboration Core Components. See "Setting Up Web Conferencing for Internet Access" on page 4-5

[2] mx is a Real-Time Collaboration internal proprietary protocol.

[3] Port will be chosen from this range.

[4] redirect is a Real-Time Collaboration internal proprietary protocol.

For deployments that are accessible from the Internet, it is enough for Internet- or extranet-facing firewalls of the DMZ to have just the traditional ports (443 and 80) open.

User Management

Oracle Real-Time Collaboration uses the Oracle Internet Directory store, which uses LDAP (Lightweight Directory Access Protocol), to authenticate its users. Any Oracle Internet Directory user can use Real-Time Collaboration. Users are created using the standard mechanisms available through Oracle Internet Directory.

See Also: *Oracle Internet Directory Administrator's Guide* for details

Real-Time Collaboration users can be assigned different roles. Roles determine the Oracle Web Conferencing functionality to which a user has access. There are three roles in Web Conferencing:

End-User Role

The end-user role, *enduser*, is the default role given to any user who logs in to the system for the first time. This role is intended for all regular users of Oracle Web Conferencing.

Business Monitor Role

The business monitor role, *businessmonitor*, is intended for those Oracle Web Conferencing users who want to monitor the system and have access to various reports that can be run on the system. Users with this role have access to the Monitor and Reports tabs, in addition to all end-user tabs in the Oracle Web Conferencing Application.

Business Administrator Role

The business administrator role, *businessadmin*, is intended for those Web Conferencing users who are in charge of administering the Web Conferencing deployment. This includes users who are responsible for supporting the end-users. Users who have this responsibility have access to the Site Management and the System Configuration tabs in the Web Conferencing Application.

Use the `imtctl` command, `modifyRole`, to assign roles to Web Conferencing users. See Chapter 10, "imtctl Command Line Utility" for more information about `imtctl`.

Web Conferencing Sites

Oracle Web Conferencing lets you create individual sites for your different lines of business (for example, sales and support) and to customize system, application, and conference level properties for those sites. Users with businessadmin privileges as described in "User Management" on page 2-10 can configure sites with `imtctl` and access the Sites tab to monitor sites. See Chapter 9, "Web Conferencing Sites" for details about creating and using sites.

Reports

Oracle Real-Time Collaboration provides various reporting capabilities, including e-mailed reports and usage trend information available within the Oracle Web Conferencing Application. Some aspects of these features require post-installation configuration, such as including sender's and receiver's e-mail addresses. See Chapter 4, "Post-Installation" for details about the properties that must be configured.

Anyone with the businessadmin or businessmonitor role as described in "User Management" on page 2-10 can see the Reports tab to view reports. See Chapter 8, "Reports" for details about Web Conferencing reports.

3

Planning for Deployment

This chapter highlights the factors to consider when deploying Oracle Web Conferencing. The issues range from sizing guidelines to firewall issues to topology considerations for an enterprise that is geographically distributed.

Sizing Guidelines

The *Oracle Web Conferencing Sizing Guide* helps you determine the hardware required to set up Web Conferencing for your specific needs. Please read this guide prior to installing and deploying Web Conferencing

Internet/Intranet Considerations

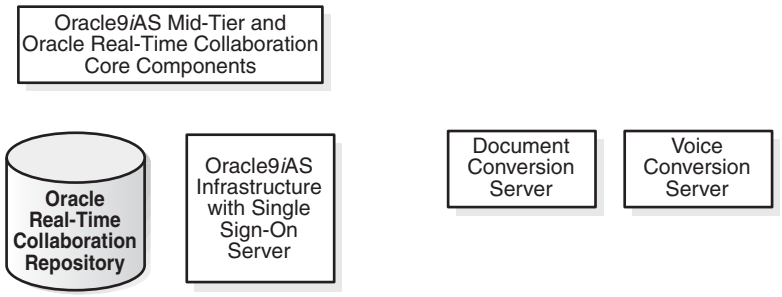
As discussed in previous chapters, the Oracle Web Conferencing system includes the following components:

- Oracle Real-Time Collaboration Core components that provide the core functionality of Web Conferencing
- The Document Conversion Server for converting Microsoft Office documents that need to be shared during a conference
- The Voice Conversion Server for streaming voice data during a conference
- The set of Oracle Real-Time Collaboration database schemas residing in an Oracle9*i*AS database

In addition, the Oracle Web Conferencing system interacts with an Oracle9*i*AS Infrastructure system to manage user sign-on and to synchronize user information with the Oracle Internet Directory.

Figure 3–1, "Oracle Real-Time Collaboration Basic Components" shows the components needed for Web Conferencing. After you have installed Web Conferencing, you will have the following components:

Figure 3–1 Oracle Real-Time Collaboration Basic Components



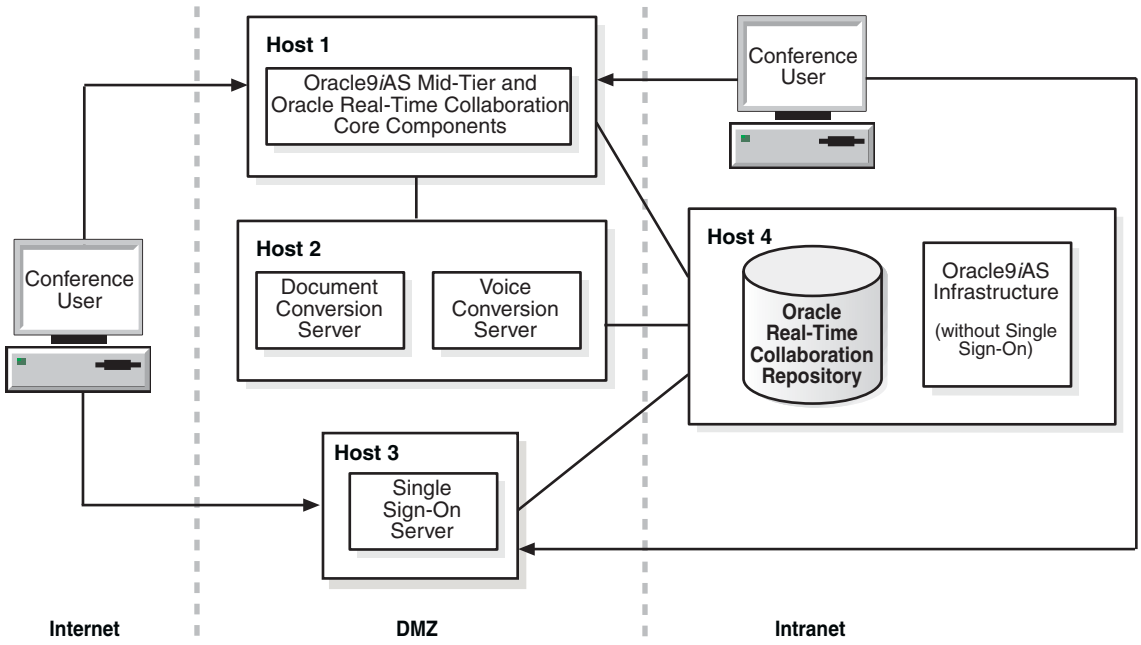
The Oracle Real-Time Collaboration Core components and Document and Voice Conversion servers are installed on Oracle9*i*AS mid-tier systems. Document and Voice Conversion servers must be installed on a computer running Microsoft Windows and Microsoft Office. (The Voice Conversion and Document Conversion servers have additional hardware and software requirements discussed in the *Oracle Web Conferencing Sizing Guide*). The supporting database and infrastructure systems are often installed on a separate database host, although they can also be installed together with the core components.

Some considerations need to be made, however, to support conference services on a corporate intranet that generally includes a secure intranet area behind a firewall, a DMZ ("demilitarized zone") between the firewall and the public Internet, and web conferencing users both within the intranet and outside the firewall in the Internet.

Figure 3–2, "Oracle Real-Time Collaboration Example Deployment" shows an example of a deployment that can allow access to users outside the corporate firewall. The components are placed as follows in the DMZ or behind the firewall:

- The Real-Time Collaboration Core Components are in the DMZ, listening on port 80 for HTTP and port 443 for HTTPS
- The Single Sign-On Server used to authenticate users entering a conference is in the DMZ, to communicate with users outside the firewall
- The Real-Time Collaboration Repository, without Single Sign-On, is behind the inner firewall
- The Oracle9*i*AS Infrastructure is inside the inner firewall

Figure 3–2 Oracle Real-Time Collaboration Example Deployment



The Single Sign-On Server and Oracle Real-Time Collaboration Core components can be located on the same mid-tier system, if desired. The Single Sign-On server can remain with the Oracle9*i*AS infrastructure if both are placed in the DMZ to serve Internet users. The repository and Oracle9*i*AS infrastructure can be on separate machines or combined, or can even be combined on a machine with the Oracle Real-Time Collaboration core components.

In addition, you must consider where the multiplexer listening ports are placed:

- *Direct TCP/IP connection to mx ports available*: All intranet users should be able to make a direct TCP/IP connection to the Real-Time Collaboration Core Components (the ports that the mx is listening on).
- *Open intra-DMZ firewalls to mx ports if necessary to support Document and Voice Conversion servers:* It is mandatory that Document and Voice Conversion Servers be able to connect to the Real-Time Collaboration Core Components they are servicing. If they are deployed in different DMZs, intra-DMZ firewalls need to be opened up on the port(s) to which the mx component on the Real-Time Collaboration Core Components machine is listening.

Load Balancer Considerations

You may use a Load Balancer (LBR) to manage processes handled by your Oracle mid-tier servers. If so, then keep these considerations in mind:

- All machines behind the LBR should have intranet-routable IP addresses and must be directly accessible from the Internet at least on the standard HTTPS port (443).
- If you use geographic load balancers – that is, load balancers to separate loads between geographic locations – then you must create Oracle Real-Time Collaboration *clusters* to partition the system based on geographical distribution (See "Real-Time Collaboration Clusters" on page 3-5).

See Chapter 6, "Sample Deployments" for an example.

Distributed Deployment Considerations

If you have multiple corporate locations that will use Web Conferencing, then you must consider:

- The number of conferences held in a single geographic location
- The number of conferences held across geographic locations
- The number of streaming-voice conferences held across geographic locations
- Whether creating clusters will help distribute Real-Time Collaboration processes to distinct geographic locations

The following sections outline some rules of thumb to apply when considering where to deploy Real-Time Collaboration core components and Voice Conversion servers.

Deploying Core Components Locally or Globally

Because data from one user's desktop is distributed to other end-user desktops during Web Conferencing, end users may experience latency issues depending on their location. Oracle Web Conferencing, by itself, cannot address network latency issues. You must resolve these issues through other mechanisms.

However, you can consider where you will deploy the Real-Time Collaboration core components, depending on how many meetings are held in specific geographic locations as described the the following two scenarios.

Scenario 1: Many Meetings in Separate Geographical Locations

If your company holds many conferences where most of the attendees are in the same geographical region, then Oracle Corporation recommends deploying a set of Real-Time Collaboration Core Components instances in that geographical region.

For example, a company has multiple divisions all over the world, and each division's employees hold many conferences with each other. Occasionally they hold conferences with employees from other divisions in different locations.

In such a scenario, it does not make sense to force users to use a Real-Time Collaboration cluster in a different location, thereby causing network latency problems for users. Instead, each location can have its own set of Real-Time Collaboration Core Components instances. See "Real-Time Collaboration Clusters" on page 3-5 for discussion of how to create geographically-oriented clusters of components.

Scenario 2: Many Meetings Across Geographical Locations

If attendees of most conferences are in different regions, then having a set of Real-Time Collaboration Core Components instances in each geographical region will not help prevent latency issues.

For example, a US-based company has outsourced its sales/support organization to a site in India. If a typical conference involves a sales agent from the site in India and a customer in the US, then there deploying Real-Time Collaboration Core Components instances in the India site will not prevent a latency problem.

Deploying Voice Conversion Server Locally or Globally

You should deploy Voice Conversion Servers in areas where the servers are able to successfully dial in to all conference numbers that will be used. It may be beneficial to deploy a Voice Conversion Server in the region where most calls terminate. For example, if a significantly large number of Web Conferencing users dial out to a

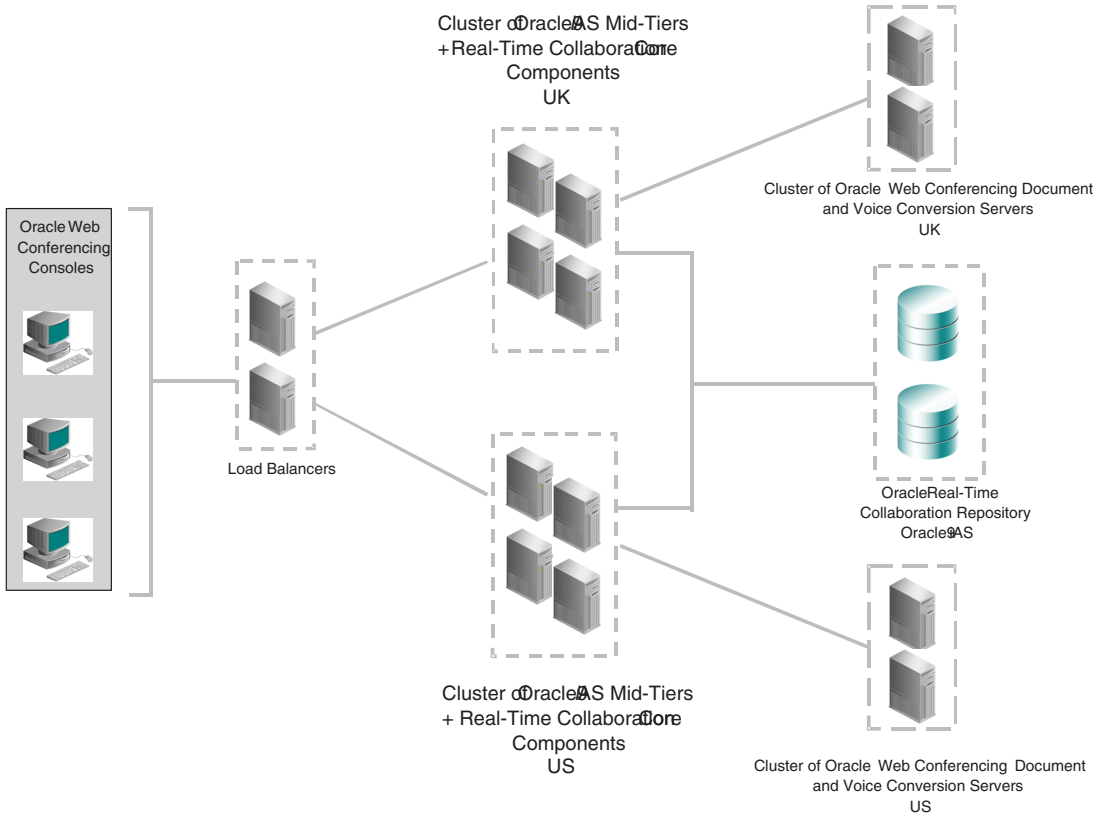
particular region like the United Kingdom, it would be beneficial to have a Voice Conversion Server deployed in the UK instead of using a Voice Conversion Server in the United States to dial the UK numbers. See "Real-Time Collaboration Clusters" for discussion of how to create geographically-oriented components.

Real-Time Collaboration Clusters

Without explicit partitioning of the instances in the system, all Web Conferencing Server processes in all instances are considered part of one group. Whenever a new conference is created and a Web Conferencing Server process needs to be chosen for a conference, the conference could be assigned to any one of the Web Conferencing Servers in the system. But, sometimes, it is useful to partition the system into clusters based on geographical distribution.

Figure 3–3, "Real-Time Collaboration Clustering" shows a sample Web Conferencing deployment where components are grouped according to their geographic locations.

Figure 3–3 Real-Time Collaboration Clustering



To create clusters like those shown in Figure 3–3, you use the InstanceLocation property. Instances that have the same value for the InstanceLocation property are part of a Real-Time Collaboration cluster. The advantages of creating a cluster include:

- **Load balancing**: All Web Conferencing Server processes in all instances that are part of the cluster become part of one pool of available servers, and the load is balanced intelligently between the different Web Conferencing Servers.
- **Load separation**: If hardware load balancers are also geographically distributed, users can be served by separate Real-Time Collaboration clusters that are more locally situated to the users.

- **High availability**: An instance could be down, which means that all Web Conferencing Servers in the instance are down, but other Collaboration Servers from other instances that are part of the same cluster can provide undisrupted service to users.

The Oracle Web Conferencing Application (OC4J_imeeting) is the component that picks the Web Conferencing Server process for a conference. The Oracle Web Conferencing Application knows the Web Conferencing Server process location and picks a Collaboration Server process in an instance which has the same value as its own location.

The Document Conversion server or Voice Conversion server can also be assigned an InstanceLocation property. If you have multiple Document and/or Voice Conversion servers that are installed at different geographic locations, and you used the InstanceLocation property to identify clusters of Real-Time Collaboration core components, you can also assign matching InstanceLocation properties to the Document and Voice Conversion servers. For example, as shown in Figure 3–3, "Real-Time Collaboration Clustering", each instance of the Document and Voice Conversion Servers in the United Kingdom is assigned to service a set of Real-Time Collaboration Core Components whose location attribute value is "UK." Each instance in the set of Document and Voice Conversion Servers in the United States is assigned to service a set of Real-Time Collaboration Core Components whose location attribute value is "US."

See Also: See "Properties to Configure Clusters" on page 5-19 for details about setting the InstanceLocation property.

4

Post-Installation

This chapter discusses post-installation steps required to get your Oracle Web Conferencingup and running. It describes:

- Post-Installation Tasks
- "Manually Starting and Stopping Web Conferencing Processes"

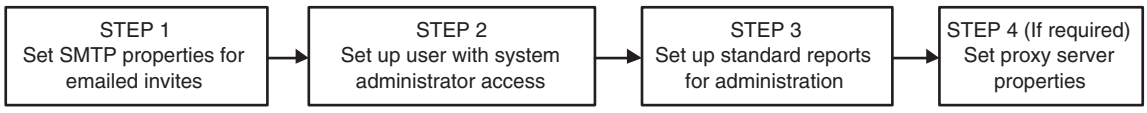
When performing the tasks described in this chapter, you will set various Real-Time Collaboration properties. For more details about properties, see Chapter 5, "Configuration". You will run imtctl commands while setting properties. For more details about the `imtctl` utility, see Chapter 10, "imtctl Command Line Utility".

Post-Installation Tasks

After you install the Oracle Web Conferencing system, there are some configuration tasks you need to take. This section outlines those steps:

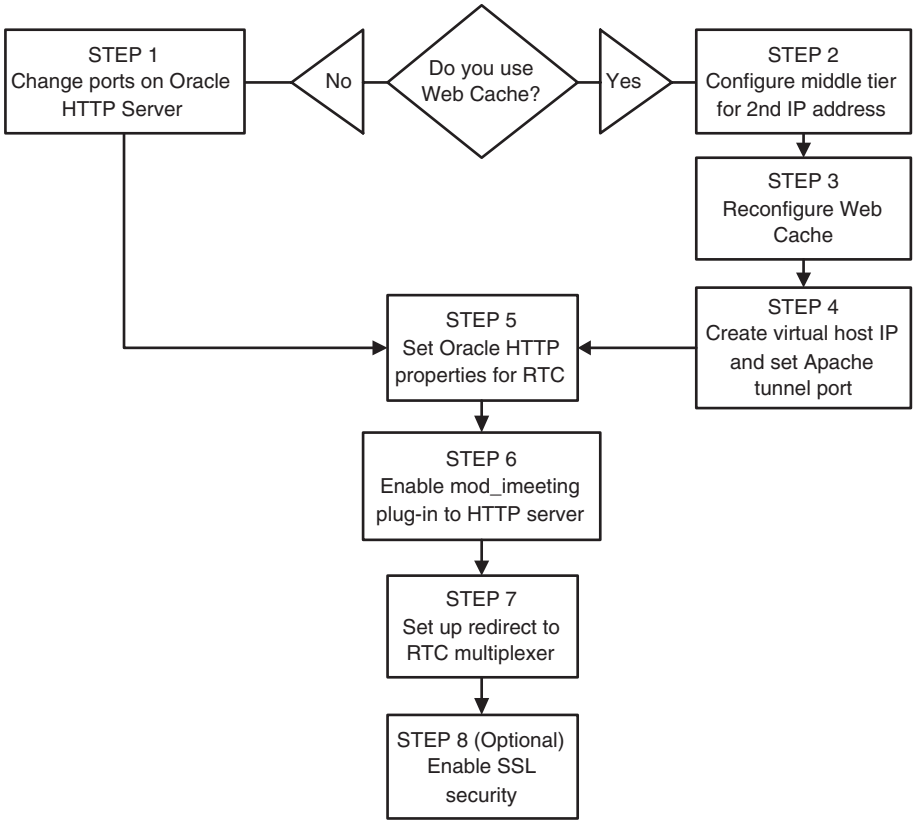
- The minimum post-configuration steps are shown in Figure 4–1. See "Setting Up Web Conferencing" on page 4-3 for an overview of these tasks.

Figure 4–1 Post-Installation Tasks for Web Conferencing



- Do you also want to hold meetings over the Internet, with meeting members outside the company firewall? Figure 4–2 shows the steps you take. Note that you *either* do step 1, or steps 2 through 4. See "Setting Up Web Conferencing for Internet Access" on page 4-5 for an overview of these tasks.

Figure 4–2 Setting Up Web Conferencing for Internet Access



- Do you want to hold conferences with streaming audio, and play back conferences with audio? See "Setting Up Document and Voice Conversion Servers for Web Conferencing" on page 4-9 for details.
- Do you want to monitor your Real-Time Collaboration components? See "Setting Service Availability Tests" on page 4-10 for details.
- You can verify the preceding configurations using the `runtests` utility. See "Verifying Configuration" on page 4-10 for details.

Types of Post-Installation Tasks

When you install the Oracle Web Conferencing system, including instances of the Oracle Web Conferencing Core components and the Document and Voice Conversion servers, you perform the tasks on both the entire web conferencing system, and on each installed instance of a Real-Time Collaboration component. You may also perform tasks on other Oracle servers as appropriate:

- **System tasks**: These tasks must be done once for the entire Oracle Web Conferencing system. In most cases, there is no need to repeat these steps when a new instance is added to the system. You can override the global settings for specific instances if necessary.
- **Instance tasks**: These must be done for every instance. The set of specific tasks that must be completed is determined by how you are deploying the system (for example, allowing Internet access or enabling SSL security).

The following sections describe the post-installation steps. Each step notes which type of task you are performing.

Setting Up Web Conferencing

You perform the following steps once so that you can use Web Conferencing to hold conferences.

Step 1 Integrate with existing e-mail. (System task)

The Web Conferencing Application can generate outbound e-mail for various purposes, such as inviting users to a conference. You must configure the Real-Time Collaboration system with information about your STMP host. You do this step once.

1. On a machine containing a Real-Time Collaboration core component instance, use a text editor to open the following script:

   ```
   $ORACLE_HOME/imeeting/bin/scripts/mail.imt
   ```

2. Edit the script to replace default values for the following Real-Time Collaboration properties with actual values for your system:

   - `SmtpHost`: The hostname of your e-mail SMTP server, for example `mailhost.mycompany.com`
   - `SmtpPort`: The port at which your e-mail SMTP server listens for mail (default is 25)
   - `EmailEnabled`: Enables the sending of e-mailed conference invitations; set this property to `true`
   - `AdminEmail`: The e-mail account name that should receive administrative e-mails from the Web Conferencing system, for example `joe.smith@oracle.com`

   See "Properties to Configure E-mail Invites" on page 5-20 for more details about these properties.

3. Set properties by executing the following script in `imtctl`:

   ```
   $ORACLE_HOME/imeeting/bin/imtctl < scripts/mail.imt
   ```

The script sets e-mail properties for your entire Real-Time Collaboration system.

Step 2 Set up the administrator role. (System task)

You must have at least one user with full business administrator privileges. The business administrator can monitor the system, view reports, create and maintain any custom sites, and check system status. To set this user, you use the `modifyRole` command within `imtctl`. You do this step once.

- On a machine containing a Real-Time Collaboration instance, enter the following command. Substitute the appropriate username for the sample shown.

  ```
  $ORACLE_HOME/imeeting/bin/imtctl modifyRole -username joe.smith@mycompany.com
  -rolename businessadmin
  ```

See "Property to Set User Privileges" on page 5-16 for more details about this command.

Step 3 Set up reports. (System task)

The system usage reports shown under the **Reports** tab are generated by a script named `imtreport` on UNIX and Linux, and `imtreport.cmd` on Microsoft Windows. The script does the following:

- Populates the Usage Report summary tables.

- Generates the report into an html file.
- Sends an e-mail to the administrator.

You should set up a cron job (on UNIX or Linux) or Scheduled Task (on Windows) to run the script on a regular basis. You set this up once.

1. On a machine containing a Real-Time Collaboration core component instance, use a text editor to open the following script:

   ```
   $ORACLE_HOME/imeeting/bin/scripts/imtreport
   ```

   On Windows, the script name is `imtreport.cmd`

2. Edit the script to replace default values for the recipient list and to choose the reports being run. The script contains comments on all of the options provided. Make sure you edit the following items:
   - Change the `export IMT_USE_BI_CLASSES` line in `imtreport` to `export IMT_USE_REPORT_CLASSES`. Failure to edit this line results in a `NoClassDefFound` error
   - Set the `SITE_ID` parameter to a specific site ID or `100` for all sites. Setting it to an empty string results in the following error: `TODAYSDATE is an invalid identifier`
   - Remove any spaces after the commas in the comma-separated list of recipients. Failure to do so results in the e-mailing of the report to the first person in the list of recipients only.
   - Set the `DISPLAY` variable at the top of `imtjvm`. Failure to do so results in the following error: `Problem with constructor javax.swing.plaf.FontUIResource...`
3. Set a cron job or Scheduled Task to execute the script in `imtctl` on a regular basis. Information in the report is broken down by week (Sunday through Saturday). Oracle recommends you run this script every Sunday.
   - On UNIX and Linux, create a cron job to run the following command:

     ```
     $ORACLE_HOME/imeeting/bin/imtctl < scripts/imtreport
     ```

   - On Windows, create a Scheduled Task to run the following command:

     ```
     %ORACLE_HOME%\imeeting\bin\imtctl < scripts/imtreport.cmd
     ```

4. Set the Web Conferencing property that defines the company name that appears at the top of all reports:

   ```
   $ORACLE_HOME/bin/imeeting/imtctl
   imtctl> setProperty -system true -pname ReportEnvironmentName -pvalue "My
   Company's Report Title"
   imtctl> exit
   ```

Step 4 Integrate with existing proxy servers. (System task)

The Web Conferencing Cobrowse mode accesses Web sites using the HTTP/S protocols from a Real-Time Collaboration core component instance. If your company uses proxy servers to control access to websites, then the Real-Time Collaboration core component may need to go through that proxy server to access websites to support Cobrowse mode during a conference (users browsing websites in a conference). You must configure the Web Conferencing system with information about your proxy server. You do this step once.

1. On a machine containing a Real-Time Collaboration core component instance, use a text editor to open the following script:

   ```
   $ORACLE_HOME/imeeting/bin/scripts/proxy.imt
   ```

2. Edit the script to replace default values for the following Real-Time Collaboration properties with actual values for your system:

   - `UserAgentProxyHost`: The hostname of your HTTP proxy.
   - `UserAgentProxySSLHost`: The hostname of your HTTPS proxy.
   - `UserAgentProxyPort`: The port for your HTTP proxy (if it is not 80)
   - `UserAgentProxySSLPort`: The port for your HTTPS proxy (if it is not 443)
   - `UserAgentProxyExclusions`: A list of URLs that should *not* be proxied, in the form: `["`*`domain1`*`", "`*`domain2`*`"]`
   - `UserAgentProxyEnabled`: Set to `true`

   See "Properties for Proxy Servers" on page 5-8 for more details about these properties.

3. Execute the script in `imtctl`:

   ```
   $ORACLE_HOME/imeeting/bin/imtctl < scripts/proxy.imt
   ```

Setting Up Web Conferencing for Internet Access

If you want to hold web conferences with users outside your company firewall, follow the steps in this section. Many of these steps need to be done on *each* of your Real-Time Collaboration core component instances.

> **Note:** Whether you choose Step 1 or Steps 2 through 4 depends on whether you have Oracle9*i*AS Web Cache turned on or off on your Real-Time Collaboration core component instances.

Step 1 If you do not use Web Cache, change ports on the Oracle HTTP Server. (System task)

If you do not use Web Cache, you must change the ports on the Oracle HTTP Server. These ports must be set to the standard values:

- Set the HTTP port to 80.
- Set the HTTPS port to 443.

After you have completed this task, you may skip steps 2 through 4 and go on to step 5.

Step 2 If you use Web Cache, configure your middle tier to use a second IP address (System task)

If you use Oracle9*i*AS Web Cache on the middle tier containing the Real-Time Collaboration core component instance, you must configure a second IP address on that middle tier.

1. On your Oracle HTTP Server, open the Apache configuration file:

   ```
   $ORACLE_HOME/Apache/Apache/conf/httpd.conf
   ```

2. Add the following line, setting the new IP address and a listening port of 443. You may optionally enter a *ServerName* if you have created an alias for this virtual host. The *wallet-file-location* shown should be set for the same wallet as the wallet used for SSL on the primary IP address.

```
<VirtualHost second-IP-address:443>
     ServerName
     Port 443
     SSLEngine on
     SSLCipherSuite SSL_RSA_WITH_RC4_128_MD5:SSL_RSA_WITH_RC4_128_SHA:SSL_RSA_
WITH_3DES_EDE_CBC_SHA:SSL_RSA_WITH_DES_CBC_SHA:SSL_RSA_EXPORT_WITH_RC4_40_
MD5:SSL_RSA_EXPORT_WITH_DES40_CBC_SHA
     SSLWallet wallet-file-location
     Listen second-ip-address:443
</VirtualHost>
```

3. Make sure the primary VirtualHost entry (the other `<VirtualHost>` entry in the same configuration file) has a Listen entry as follows:

```
<VirtualHost ....
     ...
     Listen primary-ip-address:443
</VirtualHost>
```

4. Update and then restart the Oracle HTTP Server:

```
$ORACLE_HOME/dcm/bin/dcmctl updateConfig -ct ohs -v
$ORACLE_HOME/dcm/bin/dcmctl restart -ct ohs -v
```

Step 3 If you use Web Cache, reconfigure it to listen to primary IP ports (System task)

If the Oracle9*i*AS Web Cache is bound to all network interfaces, you must reconfigure it to bind only to the primary IP address on port 443.

Step 4 If you use Web Cache, create a second IP address and set tunnel properties. (Instance task)

If you use Oracle9*i*AS Web Cache, you must create set ApacheTunnel properties on the Real-Time Collaboration core component instance to use the second IP address and port you set previously. You must repeat these steps on each Real-Time Collaboration core component instance that uses Web Cache.

1. On each machine with a Real-Time Collaboration core component, start imtctl:

```
$ORACLE_HOME/imeeting/bin/imtctl
imtctl>
```

2. Set the ApacheTunnel properties to use the second IP address and port you created earlier:

```
imtctl> setProperty -pname ApacheTunnelHost -pvalue second-IP-address
imtctl> setProperty -pname ApacheTunnelPort -pvalue 443
imtctl> exit
```

Step 5 Set Web Conferencing to use the Oracle HTTP Server settings. (Instance task)

The Real-Time Collaboration installer assumes that clients running web conferences use the actual host name configured for the Real-Time Collaboration core components machine to access collaboration resources. This might not be true in some deployments where DNS aliasing and other advanced network deployment scenarios

might require you to customize the names used to access the core component machine(s) from external locations. You set the host name and related properties using the following script on *each* Real-Time Collaboration core component machine.

1. On each machine containing a Real-Time Collaboration core component instance, use a text editor to open the following script:

   ```
   $ORACLE_HOME/imeeting/bin/scripts/hostname.imt
   ```

2. Edit the script to replace default values for the following Real-Time Collaboration properties with actual values for your system:

   - `ApacheWebHost`: The hostname of your Oracle HTTP Server.
   - `ApacheWebPort`: The port on which the Oracle HTTP Server is listening for HTTP requests, normally 80.
   - `ApacheWebSecurePort`: The port on which the Oracle HTTP Server is listening for HTTPS requests, normally 443.
   - `ApacheProtocolSecure`: If the Oracle HTTP Server is using SSL secure connections, set to `true`; if SSL is not used, set to `false`.

   See "Properties to Synchronize with Oracle HTTP Server Settings" on page 5-11 for more details about these properties.

3. Execute the script in `imtctl`:

   ```
   $ORACLE_HOME/imeeting/bin/imtctl < scripts/hostname.imt
   ```

Step 6 Enable mod_imeeting. (Server task)

As discussed in Chapter 2, "Understanding Oracle Web Conferencing", `mod_imeeting` is the Real-Time Collaboration plug-in to the Oracle HTTP Server. It lets Oracle Web Conferencing use the Oracle HTTP Server to listen for end-user connection requests. You must start `mod_imeeting` on your Oracle HTTP Server.

1. On your Oracle HTTP Server, use a text editor to open the following file:

   ```
   $ORACLE_HOME/Apache/Apache/conf/oracle_apache.conf
   ```

2. Add the following line for `mod_imeeting.conf` in `oracle_apache.conf`:

   ```
   include "$ORACLE_HOME/imeeting/conf/mod_imeeting.conf"
   ```

3. Update and then restart the Oracle HTTP Server:

   ```
   $ORACLE_HOME/dcm/bin/dcmctl  updateConfig -ct ohs -v
   $ORACLE_HOME/dcm/bin/dcmctl restart -ct ohs -v
   ```

Step 7 Set up the redirect from mod_imeeting to mx. (Instance task)

As discussed in Chapter 2, "Understanding Oracle Web Conferencing", the Oracle HTTP Server hands connections off from `mod_imeeting` to the Web Conferencing multiplexer (mx). You must follow the next steps for *each* Real-Time Collaboration core component.

1. On each machine containing a Real-Time Collaboration core component instance, use a text editor to open the following script:

   ```
   $ORACLE_HOME/imeeting/bin/scripts/redirect.imt
   ```

2. Edit the script to set the following properties:

- `MxRedirectPort`: The port (on Windows/NT) or domain name socket (on UNIX or Linux) to which the Oracle HTTP Server mod_imeeting process will connect with the mx process on this machine. Replace the default value with an actual value for your system. Enter the value in the format `"[\"NNNN\"]"`, where *NNNN* is any integer within the range 2400-49151. Be sure to enter all the double-quotes, slashes, and square brackets. Example:

  ```
  setProperty -ct mxcomm -pname MxRedirectPort -pvalue "[\"2420\"]"
  ```

- `MxRedirectEnabled`: Indicates that `MxRedirectPort` has been set. This property must be added to the script (it is not in the current `redirect.imt` script) and set to true, as follows:

  ```
  setProperty -ct mxcomm -pname MxRedirectEnabled -pvalue "true"
  ```

See "Properties to Integrate with the Oracle HTTP Server" on page 5-10 for more details.

3. Execute the script in `imtctl`:

   ```
   $ORACLE_HOME/imeeting/bin/imtctl < scripts/redirect.imt
   ```

Step 8 Set up SSL Security. (OPTIONAL System, and Instance task)

You can set Web Conferencing so that all conferences use secure HTTPS connections. To do so, you must have followed the previous steps in this section. Then you update the Oracle HTTP Server and set security properties for the Web Conferencing system, and for each Real-Time Collaboration core component instance.

1. On your Oracle HTTP Server, use a text editor to open the following file:

   ```
   $ORACLE_HOME/Apache/Apache/conf/mod_osso.conf
   ```

2. Add the following line for imt_mod_osso.conf in mod_osso.conf:

   ```
   include "$ORACLE_HOME/imeeting/conf/imt_mod_osso.conf"
   ```

3. Update and then restart the Oracle HTTP Server:

   ```
   $ORACLE_HOME/dcm/bin/dcmctl updateConfig -ct ohs -v
   $ORACLE_HOME/dcm/bin/dcmctl restart -ct ohs -v
   ```

4. The Oracle HTTP Server wallet must be set to "Auto Login mode." Use the Oracle Wallet Manager to set the wallet to this mode. See the *Oracle 9i Application Server Administrator's Guide* for more information about how to use Oracle Wallet Manager.

After you have set the Oracle HTTP Server, do the next steps on one of your Real-Time Collaboration core component instances. You only need to do these steps once.

1. On a Real-Time Collaboration core components instance, start imtctl:

   ```
   $ORACLE_HOME/imeeting/bin/imtctl
   imtctl>
   ```

2. Set the following properties as instructed, then exit imtctl.

   ```
   imtctl> setProperty -system true -pname GlobalMeetingSSLSupportEnabled -pvalue
   true
   imtctl> setProperty -system true -pname SSLRequiredforMeetings -pvalue true
   -force true
   imtct> exit
   ```

The `-force true` option forces all meetings to be secure. If you want to let users choose between secure or non-secure meetings, omit the `-force true` option.

The multiplexer (mx) on the Real-Time Collaboration core components machine uses the same wallet that the Oracle HTTP Server uses. You must set the MxWalletLocation property to identify where the wallet file is located. Do the following steps on *each* Real-Time Collaboration core components instance.

1. On each Real-Time Collaboration core components instance, start imtctl:

```
$ORACLE_HOME/imeeting/bin/imtctl
imtctl>
```

2. Set the following property as instructed, then exit imtctl.

```
imtctl> setProperty -pname MxWalletLocation -pvalue path/wallet-file
imtctl> exit
```

Setting Up Document and Voice Conversion Servers for Web Conferencing

If you set multiple InstanceLocations for Real-Time Collaboration core components to create geographical clusters, as discussed in "Distributed Deployment Considerations" on page 3-4, you will need to identify matching locations for your Document and Voice Conversion servers. There are also some additional steps you take to configure Voice Conversion servers. This section describes all of these steps.

1. If you have different `InstanceLocations` for Real-Time Collaboration core components, configure Document and Voice Conversion servers to have matching `InstanceLocations`. For example, if the location for your Real-Time Collaboration components is US, enter the following on your Voice and Conversion servers:

```
$ORACLE_HOME/imeeting/bin/imtctl setProperty -pname InstanceLocation -pvalue
"[\"US"]"
```

   See "Properties to Configure Clusters" on page 5-19 for more details.

2. Set the `VoiceDialInPrefix` property. This property provides the dial-in prefix for any voice calls made. For example, many companies require users to enter a 9 before dialing out; you can enter that value into the dial-in prefix property. On the Voice Conversion server, you would enter:

```
$ORACLE_HOME/imeeting/bin/imtctl setProperty -i instance-name -pname
"VoiceDialinPrefix" -pvalue "9"
```

   See "Property to Configure Voice Conversion Servers" on page 5-32 for details about this property.

3. Set up system dial-ins.

   System dial-ins are system-wide dial-ins that an administrator creates. They are available to all Oracle Web Conferencing users. Typically, system administrators create system dial-ins for phone conference vendors that are used commonly by the users of the system. System dial-ins serve as templates for users to create their own dial-ins; however, users cannot edit them. In many cases, system dial-ins are incomplete dialing sequences in that they contain information that is common to all users and indicate the information a user must add to make it a usable dial-in. Users should make the recommended addition to the system dial-in and save it as a user-defined dial-in for use in a conference.

   For example, phone conference vendors typically provide a PIN or conference ID to its users. A system administrator might create a system wide dial-in for the

particular conference vendor that contains the toll-free number for the vendor, followed by a placeholder for the PIN/Conference ID and the required separators like ',', '#' or '*'. Users would then modify the system dial-in by adding their PIN/Conference ID and then saving it under a different name.

Table 4–1 Dial-Ins

Dial-in Name	Dial-in Sequence
Audio Conferencing Vendor (US)	18005555555,<ADD CONF ID HERE>#,,,,,,,,,#,,,,,,,,#
Joe's account with Audio Conferencing Vendor	18005555555,8282828#,,,,,,,,, #,,,,,,,,#

System dial-ins can be managed using the `AddSysDialIn` command in `imtctl`.

```
$ORACLE_HOME/imeeting/bin/imtctl addSysDialin -name Name_for_Dial-in -sequence
dial-in sequence
```

See "Setting Conference Dial-In Information" on page 10-11 for more information about this command.

Setting Service Availability Tests

You can set up a monitoring infrastructure to periodically ping the Real-Time Collaboration Core Components. For example, assuming the URL for the instance (ApacheWebHost) is `my.company.com`, you can set a cron job (on UNIX or Linux) to ping the following URLs.

- Conference service availability—check if the machine with Real-Time Collaboration Core Components is capable of hosting a conference.

  ```
  http://my.company.com/imtapp/servlet/ImtTestServlet?mtgtest=true
  ```

- Voice Conversion Server availability—check if the machine with the Real-Time Collaboration Core Components can provide live voice streaming during a conference.

  ```
  http://my.company.com/imtapp/servlet/ImtTestServlet?voiceconvtest=true
  ```

- Document Conversion Server availability—check if this Core Components machine can successfully handle document conversion requests.

  ```
  http://my.company.com/imtapp/servlet/ImtTestServlet?docconvtest=true
  ```

See Chapter 7, "Monitoring" for more details.

Verifying Configuration

Invoke the following test to verify that the Real-Time Collaboration Core Components have been configured correctly.

```
imtctl> runtests -system true
```

To verify that any Voice Conversion Server is set up correctly, shut down the other Voice Conversion Servers, if any, that are assigned to serve the same Real-Time Collaboration Core Components to which the new server is assigned. Then, on a Real-Time Collaboration Core Components instance machine enter:

```
imtctl> runtests -testlist voiceconvtest
```

Do the same for the Document Conversion Server:

```
imtctl> runtests -testlist docconvtest
```

See Chapter 10, "imtctl Command Line Utility" for more information about the `runtests` command and what tests it runs. If any test fails, verify that the preceding post-installation steps have been done correctly.

> **Note:** If you use clusters as discussed in "Real-Time Collaboration Clusters" on page 3-5, the Voice Conversion Server availability test and the Document Conversion Server availability test will fail on the core components instance if the Document and Voice Conversion Servers do not have InstanceLocation properties that match the InstanceLocation of the Real-Time Collaboration core components instance. See "Properties to Configure Clusters" on page 5-19 for details.

Manually Starting and Stopping Web Conferencing Processes

The Real-Time Collaboration components and the additional Oracle components used by Oracle Web Conferencing are started during installation. If for some reason you need to start or stop these processes, this section describes how to do so. It covers the following topics:

- Starting and Stopping Oracle9iAS Components for Web Conferencing
- Starting or Stopping an Oracle Real-Time Collaboration Instance

This section discusses how to use the `dcmctl` and `imtctl` utilities . For more details about these commands, see the following books:

> **See Also:** *Oracle9i Application Server Administrator's Guide* for details about dcmctl

> **See Also:** Chapter 10, "imtctl Command Line Utility" for details about imtctl

Starting and Stopping Oracle9*i*AS Components for Web Conferencing

The Oracle Web Conferencing system uses several Oracle9*i*AS components such as the Oracle Web Conferencing J2EE Application (OC4J_imeeting), Oracle HTTP Server, and Oracle Process Management and Notification. This section shows you how to start or stop these processes, if necessary.

1. To determine the status of the processes, use the `dcmctl getState` command on all servers where the Real-Time Collaboration Core Components are installed:

   ```
   $ORACLE_HOME/dcm/bin/dcmctl getState -v
   ```

2. To start `OC4J_imeeting`, enter the following command:

   ```
   $ORACLE_HOME/dcm/bin/dcmctl start -co OC4J_imeeting -t 120 -v
   ```

3. To start the Oracle HTTP server, enter:

   ```
   $ORACLE_HOME/dcm/bin/dcmctl start -ct ohs -t120 -v
   ```

4. To start the Oracle Process Monitor and Notification system, enter:

```
$ORACLE_HOME/dcm/bin/dcmctl start -ct opmn
```

5. To stop any of the processes, enter the same commands but substitute `stop` for `start`.

Starting or Stopping an Oracle Real-Time Collaboration Instance

At installation time, all the Real-Time Collaboration processes are automatically started by the installer. At any other time, all Web Conferencing components can be started or stopped by using the `imtctl` utility on each instance where Web Conferencing components have been installed.

1. To check the status of Oracle Real-Time Collaboration components, use the `imtctl getState` command:

   ```
   $ORACLE_HOME/imeeting/bin/imtctl getState
   ```

2. To start any Real-Time Collaboration instance, enter the following `start` command on each machine on which an instance has been installed:

   ```
   $ORACLE_HOME/imeeting/bin/imtctl start
   ```

   The Document or Voice Conversion Servers, used to support conversion of documents for document sharing and streaming voice during conferences, are installed on Windows machines. The command to start these servers is:

   ```
   %ORACLE_HOME%\imeeting\bin\imtctl start
   ```

3. You can stop any Web Conferencing component by using the `stop` command:

   ```
   $ORACLE_HOME/imeeting/bin/imtctl stop
   ```

For more infromation about `imtcl` commands and syntax, see Chapter 10, "imtctl Command Line Utility".

5

Configuration

This chapter describes configuration of the Web Conferencing system in detail, including both required and optional configuration tasks. This chapter also explains the basics of the property management system and configuration management, and lists the available properties.

When you add a new instance to the system, follow the steps in Chapter 4, "Post-Installation". Use this chapter as a reference for all configuration details.

Using Web Conferencing Properties

You use properties to configure the Web Conferencing system. You use the `imtctl` utility to set properties. This chapter gives the syntax for setting various properties with the `imtctl` utility; for more details about using the utility, see Chapter 10, "imtctl Command Line Utility". The following sections discuss:

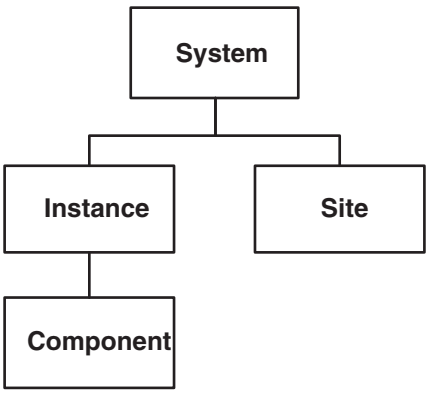
- "How Properties Affect Web Conferencing"
- "Multiple-Scope Properties"
- "Listing Current Property Settings"

How Properties Affect Web Conferencing

You set properties to customize attributes of the entire system, as well as aspects of individual instances and components. You can also set properties for particular sites that you create, to control conference attributes for a particular line of business or department within your company (see Chapter 9, "Web Conferencing Sites" for information about creating sites).

The portion of the Web Conferencing system that a property affects is called its *scope*. Figure 5–1, "Hierarchy of Configuration Properties" shows the scopes of the properties and their relationship to each other.

Figure 5–1 Hierarchy of Configuration Properties



- **System properties**: Set for an entire Web Conferencing system. They are typically set once and do not need to be changed when a new instance or new set of Real-Time Collaboration Core Components is added to the system.

  Example: Load balancer (LBR) properties. If an enterprise has one LBR, its properties usually only need to be set once.

- **Instance properties**: Set on a single instance. Any change in the value of these properties affects that instance only.

  Example: Location properties. Each instance in the system may have its location attribute set to a different value.

- **Component properties**: Set on a component in a specific instance. A change in the properties affects the configuration of that component only.

  Example: Properties to control the number of processes for a component. Each component in an instance can have a different number of processes. For example, the multiplexer (mx) component could have two mx processes, and the Web Conferencing Server component could have four Collaboration Server processes.

  Note that changes to component properties apply to that component in that instance only. Changes do not affect the same components in other instances. Therefore, if, for example, you want to set the number of mx processes to five on all instances, you must set this property for each instance.

- **Site properties**: Set on a specific site, to configure web conferencing for a particular line or business or department. See Chapter 9, "Web Conferencing Sites" for details about creating sites.

  Example: The PreLoginMessage property. This property controls the text of a message that appears under the login link on the login page; you could have a different message for each site you create.

Multiple-Scope Properties

Some properties affect multiple scopes. Oracle Real-Time Collaboration provides the following features for properties:

- I**nheritance**: The property value at a particular scope can be inherited from higher up in the hierarchy. For example, if you set the log level at the system scope, all instances and components in instances will use that log level.

  The path of inheritance follows the order shown, from top to bottom, in Figure 5–1, "Hierarchy of Configuration Properties".

- **Allow Override**: A property can be set at a certain scope, but the value can be overridden lower down in the hierarchy and set to a different value. For example, the log level could be set at the system scope, but you can change the log level of a specific instance. All other instances will maintain the log level set at the system scope.

This approach provides an administrator maximum flexibility and ease in configuring the system. You set a property's scope by including a scope option with the `imtctl` command, for example, `-system` or `-instance`. See Chapter 10, "imtctl Command Line Utility" for more details about imtctl.

The rest of the sections in this chapter describe how to use individual properties. For each property, the scope(s) that apply are noted as shown in the left column of the following table:

Table 5–1 Available Scopes for Properties

Scope	Property Affects:
system	Property can be set to affect the entire system
instance	Property can be set to affect an instance
component	Property can be set to affect a component
system and instance	Property can be set to affect the entire system, but can be overridden for a specific instance
instance and component	Property can be set to affect an instance, but can be overriden for a specific component in that instance
system, instance, component	Property can be set to affect the entire system, but can be overridden for a specific instance, and in turn overridden for a specific component in that instance
site	Property can be set to affect an entire site
system and site	Property can be set to affect the entire system, but can be overridden for a specific site

Listing Current Property Settings

You can display a listing of current property settings using the imtctl command, `getProperties`. See "getProperties" on page 10-9 for more information about viewing existing property settings.

List of Web Conferencing Properties

The following table lists properties available for configuring Oracle Web Conferencing in alphabetical order. The next sections, discuss the properties in groups based on what portions of the Web Conferencing the properties configure.

Table 5–2 Web Conferencing Properties

Property	Configures	Details	Section Covered
AdditionalLocationsServed	Cluster of multiple instances	Sets locations that Document and Voice Conversion servers serve	"Properties to Configure Clusters"
AdminEmail	E-mail conference invites	Sets the administrator's e-mail for e-mail invites.	"Properties to Configure E-mail Invites"
AllowPublicMeetings	Conference defaults	Allows or prevents setting a meeting to be public	"Properties to Configure Conference Runtime Characteristics"
ApacheProtocolSecure	HTTP Server	Sets whether the Apache HTTP server is using SSL security	"Properties to Synchronize with Oracle HTTP Server Settings"
ApacheTunnelHost	Second IP Address for WebCache	Sets the second IP address for a virtual host in httpd.conf	"Properties to Configure Web Conferencing with Oracle9iAS Web Cache"
ApacheTunnelPort	Second IP Address for WebCache	Sets the port for the tunnel for a second IP address	"Properties to Configure Web Conferencing with Oracle9iAS Web Cache"
ApacheWebHost	HTTP Server	Sets the host name for the Apache server	"Properties to Synchronize with Oracle HTTP Server Settings"
ApacheWebPort	HTTP Server	Sets the port on which the HTTP (Apache) server is listening for HTTP requests	"Properties to Synchronize with Oracle HTTP Server Settings"
ApacheWebSecurePort	HTTP Server	Sets the port on which the HTTP server is listening for HTTPS requests	"Properties to Synchronize with Oracle HTTP Server Settings"
DefaultTimeZoneName	Time zone settings	Sets the time zone for this Web Conferencing system	"Property to Configure Time Zones"
EarlyJoinMinutes	Conference defaults	Sets the number of minutes before the scheduled start time that attendees can join a conference	"Properties to Configure Conference Runtime Characteristics"
EnableChat	Conference defaults	Allows or prevents using chat in conferences	"Properties to Configure Conference Runtime Characteristics"
EnableCobrowseMode	Conference defaults	Allows or prevents cobrowsing in conferences	"Properties to Configure Conference Runtime Characteristics"
EnableDesktopSharingMode	Conference defaults	Allows or prevents desktop sharing in conferences	"Properties to Configure Conference Runtime Characteristics"
EnableDocumentPresentation Mode	Conference defaults	Allows or prevents document presentation mode in conferences	"Properties to Configure Conference Runtime Characteristics"
EnableRecording	Conference defaults	Allows or prevents recording of conferences	"Properties to Configure Conference Runtime Characteristics"
EnableSharedControl	Conference defaults	Allows or prevents shared control between a host and attendee	"Properties to Configure Conference Runtime Characteristics"

Table 5–2 (Cont.) Web Conferencing Properties

Property	Configures	Details	Section Covered
EnableWhiteboardMode	Conference defaults	Allows or prevents whiteboard mode in conferences	"Properties to Configure Conference Runtime Characteristics"
FirstAttendeeBehavior	Conference defaults	Sets the first attendee to a meeting to be co-presenter	"Properties to Configure Conference Runtime Characteristics"
GlobalMeetingSSLSupportEnabled	SSL Security	Enables SSL for Web Conferencing	"Properties to Configure SSL Security"
GlobalProtocolSecure	Load Balancer Integration	Sets SSL security on or off for the global Web host	"Properties to Integrate with a Load Balancer"
GlobalWebHost	Load Balancer Integration	Name of the global Web host for this Web Conferencing system	"Properties to Integrate with a Load Balancer"
GlobalWebPort	Load Balancer Integration	The HTTP port of the global Web host	"Properties to Integrate with a Load Balancer"
GlobalWebSecurePort	Load Balancer Integration	The HTTPS port of the global Web host	"Properties to Integrate with a Load Balancer"
InstanceLocation	Cluster of multiple instances	Sets a location for an instance	"Properties to Configure Clusters"
IsCalendarOCSInstalled	Force scheduling with Oracle Calendar	Suppresses the Schedule tab so that users must schedule web conferences using Oracle Calendar	"Suppressing the Schedule Tab for Web Calendar Users"
LogFlushInterval	Log Reports	Sets when logs are flushed	"Properties to Configure Logging"
LogLevel	Log Reports	Sets levels at which log info is recorded	"Properties to Configure Logging"
LogPath	Log Reports	Sets where logs are saved	"Properties to Configure Logging"
MeetingAllControlLayout	Conference defaults	Sets whether all attendees can control the layout of the console	"Properties to Configure Conference Runtime Characteristics"
MeetingAllUseTools	Conference defaults	Sets whether all attendees can use annotation tools	"Properties to Configure Conference Runtime Characteristics"
MeetingAllViewAttendeeList	Conference defaults	Sets whether attendees can view the list of attendees	"Properties to Configure Conference Runtime Characteristics"
MeetingAllViewStatusMsg	Conference defaults	Sets whether attendees can view status messages for a conference	"Properties to Configure Conference Runtime Characteristics"
MeetingAutoHide	Conference defaults	Sets whether the conference console should be hidden when starting a conference	"Properties to Configure Conference Runtime Characteristics"
MeetingChatInterface	Conference defaults	Sets whether the Chat interface is "docked" or a pop-up window	"Properties to Configure Conference Runtime Characteristics"
MeetingChatType	Conference defaults	Sets what type of Chat is available for a conference	"Properties to Configure Conference Runtime Characteristics"

Table 5–2 (Cont.) Web Conferencing Properties

Property	Configures	Details	Section Covered
MeetingDialInAndConfId	Conference defaults	Sets default dial-in information for any scheduled conference	"Properties to Configure Conference Runtime Characteristics"
MeetingDurationMinutes	Conference defaults	Sets the default duration for any scheduled conference	"Properties to Configure Conference Runtime Characteristics"
MeetingEmailJoin	Conference defaults	Allows attendees to join a conference directly from an e-mailed invitation to a scheduled conference	"Properties to Configure Conference Runtime Characteristics"
MeetingEndUrl	Conference defaults	Sets the ending URL for a scheduled conference	"Properties to Configure Conference Runtime Characteristics"
MeetingExpandHelpText	Conference defaults	Sets whether the Help banner appears in the console	"Properties to Configure Conference Runtime Characteristics"
MeetingExpandSecondRow	Conference defaults	Sets whether the second row of the console banner (with network status, poll, preference, and details commands) appears	"Properties to Configure Conference Runtime Characteristics"
MeetingInformation	Conference defaults	Sets additional information listed for any scheduled conference	"Properties to Configure Conference Runtime Characteristics"
MeetingIsInstant	Conference defaults	Sets whether a conference is an "instant" conference	"Properties to Configure Conference Runtime Characteristics"
MeetingPublishOption	Conference defaults	Sets whether the attendee list is published to attendees by default	"Properties to Configure Conference Runtime Characteristics"
MeetingSharedControlSendPoll	Conference defaults	Sets whether attendees sharing control with the presenter can send polls	"Properties to Configure Conference Runtime Characteristics"
MeetingSharedControlUseTools	Conference defaults	Sets whether attendees sharing control with the presenter can use annotation tools	"Properties to Configure Conference Runtime Characteristics"
MeetingShouldSendEmail	Conference defaults	Controls whether e-mail is sent to attendees by default for a scheduled conference	"Properties to Configure Conference Runtime Characteristics"
MeetingStartUrl	Conference defaults	Sets the starting URL for a scheduled conference	"Properties to Configure Conference Runtime Characteristics"
MeetingTypeOfMtg	Conference defaults	Sets the default conference type (regular, public, or restricted) for any scheduled conference	"Properties to Configure Conference Runtime Characteristics"
MxListenPort	Multiplexer (mx)	Set the ports on which the mxcomm processes listen	"Properties to Configure the Multiplexer (Mx)"
MxRedirectEnabled	Multplexer (mx) redirect	Indicates that a port or socket name has been set using MxRedirectPort	"Properties to Configure the Multiplexer (Mx)"

Table 5–2 (Cont.) Web Conferencing Properties

Property	Configures	Details	Section Covered
MxRedirectPort	Multiplexer (mx) redirect	Sets the multiplexer port or socket name for handing off from mod_imeeting to the mx	"Properties to Integrate with the Oracle HTTP Server"
MxWalletLocation	SSL Security	Oracle Wallet used by the Oracle HTTP Server	"Properties to Configure SSL Security"
MxWalletPassword	SSL Security	Set a password for the Oracle Wallet	"Properties to Configure SSL Security"
PreLoginMessage	Web Conferencing Application Pages	Customize text of a message that appears under the login link on the login page	"Properties to Configure the Application Pages"
PrivacyLink	Privacy statement	Sets the URL for the privacy link	"Customizing the Privacy Statement"
PrivacyText	Privacy statement	Sets the text of the privacy statement from the privacy link	"Customizing the Privacy Statement"
PublicMeetingNumRows	Web Conferencing Application Pages	Set the number of rows in the Public Conferences table	"Properties to Configure the Application Pages"
PublicMeetingSortBy	Web Conferencing Application Pages	Set the criteria by which conferences in the Public table are sorted	"Properties to Configure the Application Pages"
PublicMeetingSortOrder	Web Conferencing Application Pages	Control the order in which conferences in the Public Conferences table are sorted	"Properties to Configure the Application Pages"
QuicklinkNName	Web Conferencing Application Pages	Configure the name of one of the QuickLinks	"Configuring the Quicklinks Bin"
QuicklinkNURL	Web Conferencing Application Pages	Set the URL associated with one of the QuickLinks	"Configuring the Quicklinks Bin"
ReportEnvironmentName	Reports	Set the name of the environment (site, system, etc.) for a report	"Property to Configure E-Mailed Usage Reports"
ShowPrivacyLink	Privacy statement	Allows or suppresses display of the link to a privacy statement	"Customizing the Privacy Statement"
SmtpHost	E-mail conference invites	Identifies the host name of the e-mail SMTP server	"Properties to Configure E-mail Invites"
SmtpPort	E-mail conference invites	Listen port of the e-mail SMTP	"Properties to Configure E-mail Invites"
SrvNumProcs	Multiplexer (mx)	Set the number of mx processes for a component	"Properties to Configure the Multiplexer (Mx)"
SSLRequiredforMeetings	SSL Security / Conference defaults	Controls whether meetings must use SSL security.	"Properties to Configure SSL Security"
StartupMeetingMode	Conference defaults	Sets the default start mode for any conference	"Properties to Configure Conference Runtime Characteristics"
UserAgentProxyEnabled	Proxy server info	Enables all proxy settings	"Properties for Proxy Servers"
UserAgentProxyExclusions	Proxy server info	Sets a list of URLS that should not be proxied	"Properties for Proxy Servers"

Table 5–2 (Cont.) Web Conferencing Properties

Property	Configures	Details	Section Covered
UserAgentProxyHost	Proxy server info	Sets the hostname of the HTTP proxy server	"Properties for Proxy Servers"
UserAgentProxyPort	Proxy server info	Sets the port number for the proxy server	"Properties for Proxy Servers"
UserAgentProxySSLHost	Proxy server info	Sets the hostname of the HTTPS proxy server	"Properties for Proxy Servers"
UserAgentProxySSLPort	Proxy server info	Sets the port number of the SSL proxy server	"Properties for Proxy Servers"
VoiceDialInPrefix	Voice conference defaults	Sets the dial-in prefix for a Voice Conversion server	"Property to Configure Voice Conversion Servers"

Properties to Configure Ports and Network Connectivity

This section discusses all properties that pertain to configuring Real-Time Collaboration network connectivity. See "Setting Up Web Conferencing" on page 4-3 for steps to set up proxy server properties using an `imtctl` script. See "Setting Up Web Conferencing for Internet Access" on page 4-5 for steps to integrate and synchronize with the Oracle HTTP server using `imtctl` scripts. This section covers the following properties:

- "Properties for Proxy Servers"
- Properties to Integrate with the Oracle HTTP Server
- "Properties to Configure Web Conferencing with Oracle9iAS Web Cache"
- "Properties to Configure the Multiplexer (Mx)"
- "Properties to Integrate with a Load Balancer"

Properties for Proxy Servers

The following properties configure the proxy settings used by all Oracle Web Conferencing HTTP client code to do tasks like cobrowsing and cross-instance HTTP-based diagnostics. See "Setting Up Web Conferencing" on page 4-3 for steps to set up proxy server properties using an `imtctl` script.

UserAgentProxyHost

Description: Sets the host name of the HTTP proxy.

Default value: None

Valid values: host name

Scope: Component

Examples:

- To set the proxy to www-proxy.company.com for the entire system, enter:

```
imtctl> setProperty -system true -pname UserAgentProxyHost -pvalue
"www-proxy.company.com"
```

- To set the proxy to www-proxy.company.com for the current instance, enter:

```
imtctl> setProperty -pname UserAgentProxyHost -pvalue "www-proxy.company.com"
```

UserAgentProxySSLHost

Description: Sets the host name of the HTTPS proxy.

Default value: None

Valid values: valid host name

Scope: Component

Examples:

- To set the proxy to www-proxy.company.com for the entire system, enter:

```
imtctl> setProperty -system true -pname UserAgentProxySSLHost -pvalue
"www-proxy.company.com"
```

- To set the proxy to www-proxy.company.com for the current instance, enter:

```
imtctl> setProperty -pname UserAgentProxySSLHost -pvalue
"www-proxy.company.com"
```

UserAgentProxyPort

Description: Sets the proxy port.

Default value: 80

Valid values: Any port

Scope: Component

Examples:

- To set the proxy port to 80 for the entire system, enter:

```
imtctl> setProperty -system true -pname UserAgentProxyPort -pvalue 80
```

- To set the proxy port for the current instance, enter:

```
imtctl> setProperty -pname UserAgentProxyPort -pvalue 80
```

UserAgentProxySSLPort

Description: Sets the SSL proxy port.

Default value: 443

Valid values: Any port

Scope: Component

Examples:

- To set the SSL proxy port to 443 for the entire system, enter:

```
imtctl> setProperty -system true -pname UserAgentProxySSLPort -pvalue 443
```

- To set the SSL proxy port for the current instance, enter:

```
imtctl> setProperty -pname UserAgentProxySSLPort -pvalue 443
```

UserAgentProxyExclusions

Description: Sets a list of URLs for domains or hosts that should *not* be proxied.

Default value: None

Valid values: (simple tail-matching algorithm). The format of the value must be "[\"domain1\", \"domain2\"]"

Scope: System, Instance. The property can be set at system scope but can be overridden for a specific component.

Examples:

- To set the exclusions for the whole system for domains company.com and companycorp.com, enter:

```
imtctl> setProperty -system true -pname UserAgentProxyExclusions -pvalue
"[\".company.com\", \".companycorp.com\"]"
```

- To set the exclusions for the current instance for domains company.com and companycorp.com, enter:

```
imtctl> setProperty -pname UserAgentProxyExclusions -pvalue
"[\".company.com\", \".companycorp.com\"]"
```

UserAgentProxyEnabled

Description: Enables the proxy settings. None of the other proxy settings take effect unless this is set to true.

Default value: true

Valid values: true/false

Scope: Component

Example: To enable the proxy setting, enter:

```
imtctl> setProperty -system true -pname UserAgentProxyEnabled -pvalue true
```

Properties to Integrate with the Oracle HTTP Server

To integrate with the Oracle HTTP server, you **must** first enable the `mod_imeeting` process. See "Setting Up Web Conferencing for Internet Access" on page 4-5 for details about this step, and about setting the `MxRedirectPort` property.

Properties to Redirect the Multiplexer (MX) Port

The following property sets the port that redirects the connections from the `mod_imeeting` process to the multiplexer. See "Client Connection Details" on page 2-7 for more details about this hand-off.

> **Note:** You must do the tasks described in "Setting Up Web Conferencing for Internet Access" before settting the mx redirect properties.

MxRedirectPort

Description: Internet end-users connecting through the Web Conferencing Console typically connect to the Oracle HTTP server and `mod_imeeting`. Ultimately, the socket is handed off to the multiplexer by `mod_imeeting`, and this property facilitates the communication between `mod_imeeting` and the mx process on the instance.

Default value: None. The value differs depending on the platform:

- UNIX: Not a port, but a name. The socket is handed off using Domain Name Sockets.
- NT: A port.
- Linux: Not a port, but a name.

Valid Values: Integer within the range 2400-2700

Scope: Component.

Examples:

- For one mx process (installer default), enter:

```
imtctl> setProperty -ct mxcomm -pname MxRedirectPort -pvalue "[\"2420\"]"
```

- For two mx processes, enter:

```
imtctl> setProperty -ct mxcomm -pname MxRedirectPort -pvalue
"[\"2420\",\"2421\"]"
```

Typically, these two examples are enough to support all modes of connections if Oracle9iAS Web Cache is not present on the Real-Time Collaboration Core Components machine, and Oracle HTTP Server is the listener.

MxRedirectEnabled

Description: Indicates that the `MxRedirectPort` has been set and must be used by `mod_imeeting` when handing off sockets to the multiplexer. This property must be set to true if `MxRedirectPort` is used.

Default value: False.

Valid Values: True, false.

Scope: Component.

Examples:

```
imtctl> setProperty -ct mxcomm -pname MxRedirectEnabled -pvalue "true"
```

Properties to Synchronize with Oracle HTTP Server Settings

You change the following properties to synchronize with the Oracle HTTP Server settings. Again, see "Setting Up Web Conferencing for Internet Access" on page 4-5 for detailed steps to set up these properties, including how to use an `imtctl` script to set the properties.

ApacheWebHost

Description: Sets the hostname for the Apache server. The Real-Time Collaboration installer assumes that the actual host name configured for the Real-Time Collaboration Core Components machine is used by clients to access Oracle Web Conferencing resources. This might not be true in some deployments where DNS aliasing and other advanced network deployment scenarios might require customization of the names used to access Web Conferencing resources from external locations.

Default value: Actual host name derived at installation by the installer.

Valid values: Any host name

Scope: Instance

Example: If the actual host name is web10.us.oracle.com, but if the name to be used is imeeting.company.com, enter:

```
imtctl> setProperty -pname ApacheWebHost -pvalue "imeeting.company.com"
```

ApacheWebPort

Description: The port on which the Oracle HTTP Server is listening for HTTP requests

Default value: 7777

Valid values: Any legal port value

Scope: Instance

Example: If the Oracle HTTP Server on the instance is listening on port 80 instead of 7777 for HTTP requests (a typical setup), enter:

```
imtctl> setProperty -pname ApacheWebPort -pvalue 80
```

> **Note:** Setting the port to a value other than 80 may prevent access to users coming behind remote proxies.

ApacheWebSecurePort

Description: The port on which Oracle HTTP Server is listening for HTTPS requests.

Default value: 4443

Valid values: Any legal port value

Scope: Instance

Example: If the Oracle HTTP Server on the instance is listening on port 443 instead of 4443 for HTTPS requests, enter:

```
imtctl> setProperty -pname ApacheWebSecurePort -pvalue 443
```

> **Note:** Setting this to something other than 443 may prevent access to users behind remote proxies.

ApacheProtocolSecure

Description: Sets whether the ApacheWebHost is using SSL.

Default value: false

Valid values: true/false

Scope: System

Example: If the Oracle HTTP Server Web host is using SSL, this property needs to be set to true, as follows:

```
imtctl> setProperty -pname ApacheProtocolSecure -pvalue true
```

Properties to Configure Web Conferencing with Oracle9iAS Web Cache

If Oracle9iAS Web Cache is present on the Real-Time Collaboration core components machine, you must create a second IP address and a virtual host on the Oracle HTTP server. See "Setting Up Web Conferencing for Internet Access" on page 4-5 for details about how to set these items. Then you set a tunnel host with the properties listed.

Note: You must do the tasks noted in "Setting Up Web Conferencing for Internet Access" before setting the properties listed.

ApacheTunnelHost

Description: The second IP address for which a virtual host has been configured in `httpd.conf`

Default value: None

Valid values: An IP address

Scope: Instance

Example: If the second IP address is 145.35.28.4, enter:

```
imtctl> setProperty -pname ApacheTunnelHost -pvalue 145.35.28.4
```

ApacheTunnelPort

Description: Sets the tunnel port. It should be 443. Having a value other than 443 creates problems, such as preventing connection from clients from other intranets that attempt to connect through their proxy across the Internet.

Default value: 443

Valid values: A port number

Scope: Instance

Example: This is the default setting that sets the tunnel port to 443.

```
imtctl> setProperty -pname ApacheTunnelPort -pvalue 443
```

Properties to Configure the Multiplexer (Mx)

You may optionally use the following properties to set up the number of processes handled by the multiplexer (mx). The mx is the communication hub to which all component processes on an instance connect, including the Web Conferencing Server processes. Through the Web Conferencing Console, users connect to the mx processes for conferences hosted by a Web Conferencing Server on an instance. The Voice Conversion Server and the Document Conversion Server processes servicing this instance then connect to the mx.See Chapter 2, "Understanding Oracle Web Conferencing" for an overview of the multiplexer processes and how they handle Web Conferencing connection requests.

If users access web conferencing over the Internet, you can specify the port that redirects the connections from the `mod_imeeting` process to the multiplexer.

SrvNumProcs

Description: Sets the processes for each component. Setting this parameter for component type `clbsrvr` does not affect any other parameter. However, if you change this property for the mx component, you must make sure there are enough ports to match.

Default value: 1 for component type `mxcomm`, 4 for component type `clbsrv`

Valid values: Depends on the amount of memory on the system.

Scope: Component.

Examples:

- To set the number of mx processes to 2 on the current instance, enter:

```
imtctl> setProperty -ct mxcomm -pname SrvNumProcs -pvalue 2
```

- To set the number of Web Conferencing Server processes to 8 on the current instance, enter:

```
imtctl> setProperty -ct clbsvr -pname SrvNumProcs -pvalue 8
```

MxListenPort

Description: Specifies a list of ports on which the processes in the `mxcomm` component listen.

Default value: None (set during installation)

Valid values: 2400-2700

Scope: Component.

Examples:

- To set one mx communication process that listens on 2400 direct port, enter:

```
imtctl> setProperty -ct mxcomm -pname MxListenPort -pvalue "[\"2400\"]"
```

- To set two mx processes, one of which listens on 2400 direct port and the other one on 2401 on an instance, enter:

```
imtctl> setProperty -ct mxcomm -pname MxListenPort -pvalue
"[\"2400\",\"2401\"]"
```

Properties to Integrate with a Load Balancer

A load balancer (LBR) provides a single published address to the client browsers, while distributing requests to multiple Real-Time Collaboration core component middle tiers that actually serve the requests. It acts as a global Web host for all of the requests. The load balancer itself is a very fast network device that can distribute requests to a large number of physical servers.

Generally, you set the Web Conferencing properties described once for the entire system. However, if you have a different set of published addresses based on geographical considerations, each of these addresses would have multiple Real-Time Collaboration core components serving the requets. In this second scenario, you must set the following properties on each Real-Time Collaboration instance.

GlobalWebHost

Description: The name of the global Web host.

For example, there could be multiple machines behind a load balancer (server1.oracle.com, server2.oracle.com), but the Web host name you wish to appear in the URL used to join a conference is server1.oracle.com.

The name cannot work if arbitrarily DNS registered.

> **Note:** Once set, the only way to unset this property is to use the `-pvaluenull true` option with the setProperty command. See Chapter 10, "imtctl Command Line Utility" for more details about this option.

Default value: None

Valid values: A host name, such as machine.company.com or machine.us.company.com.

Scope: System, Instance

Example: To set the global Web host to server1.oracle.com, enter:

```
imtctl> setProperty -system true -pname GlobalWebHost -pvalue "server1.oracle.com"
```

GlobalWebPort

Description: The HTTP port of the global Web host.

Default value: 80

Valid values: Any legal port value

Scope: System, Instance

Example: To reset the global Web host to listen on port 80 for HTTP requests, enter:

```
imtctl> setProperty -system true -pname GlobalWebPort -pvalue 80
```

> **Note:** Setting the port to a value other than 80 may prevent access to users behind remote proxies.

GlobalWebSecurePort

Description: The HTTPS port of the global Web host.

Default value: 443

Valid values: Any legal port value

Scope: System, Instance

Example: To reset the global Web host to listen on port 443 for HTTPS requests, enter:

```
imtctl> setProperty -system true -pname GlobalWebSecurePort -pvalue 443
```

> **Note:** Setting the port to a value other than 443 may prevent access to users behind remote proxies.

GlobalProtocolSecure

Description: Sets whether the global Web host is using SSL security.

Default value: false

Valid values: false, true

Scope: System, Instance

Example: If the global Web host uses SSL, set the property to true:

```
imtctl> setProperty -system true -pname GlobalProtocolSecure -pvalue true
```

Property to Set User Privileges

Use the `imtctl` command line interface to assign roles to Web Conferencing users. Example:

```
imtctl> modifyRole -username "anyusername" -rolename "businessadmin"
```

See "Setting User Roles" on page 10-10 for details about this command.

Properties to Configure SSL Security

Oracle Real-Time Collaboration provides complete conference runtime security, as well as application security by providing complete support for HTTPS Secure Socket Layer (SSL).

In order configure SSL setup, the Web Conferencing properties that need to be in sync with Oracle HTTP Server settings and any load balancer settings to be set up, as well. The following properties need to be set:

- ApacheWebSecurePort, discussed on page 5-12
- ApacheProtocolSecure, discussed on page 5-12
- GlobalWebSecurePort, discussed on page 5-15
- GlobalProtocolSecure, discussed on page 5-15

In addition, you must update the `imt_mod_osso.conf` file on the Oracle HTTP Server as described in the final step of "Setting Up Web Conferencing for Internet Access" on page 4-5.

> **Note:** Read the SSL steps in "Setting Up Web Conferencing for Internet Access" before you set any of the properties discussed in this section.

The following sections discuss both the properties that an administrator can use to configure security options, as well as user controls for setting secure conferences and additional configuration for cobrowsing secure sites.

> **Note:** An administrator can mandate that some or all pages use HTTPS, based on Oracle9*i*AS controls. Oracle Corporation strongly recommends that an administrator make sure that the Oracle9iAS Single Sign-On access during a user login is done using HTTPS.

By default, SSL is disabled for the entire system. However, an administrator can enable the use of SSL for the entire system. Enabling SSL does *not* mean that it is mandated; it means that the conference runtime SSL is available. An additional property can be used to mandate use of SSL for all conferences in a system or at a site.

GlobalMeetingSSLSupportEnabled

Description: Enables the conference runtime SSL for the entire Real-Time Collaboration system. This means that users can choose to hold secure meetings, not that they are required to do so.

Default value: false

Valid values: false/true

Scope: System

Example: To enable SSL conferences on an entire Web Conferencing system, enter:

```
imtctl> setProperty -system true -pname GlobalMeetingSSLSupportEnabled pvalue
"true"
```

MxWalletLocation

Description: The mx uses the same wallet that the Oracle HTTP Server uses on the Real-Time Collaboration Core Components machine. The only requirement is that the wallet must be in "Auto Login mode." Use the Oracle Wallet Manager to change the wallet to "Auto Login mode." Then set the MxWalletLocation property to the location of the wallet.

> **See Also:** Oracle9i Application Server Administrator's Guide for more information on how to use Oracle Wallet Manager

Default value: Default value is unset, in which case the location `$ORACLE_HOME/Apache/Apache/conf/ssl.wlt/default` is used.

Valid values: Full path of the wallet

Scope: Instance

Example:

```
imtctl> setProperty -pname MxWalletLocation -pvalue
"/902ias/Apache/Apache/conf/ssl.wlt/default"
```

MxWalletPassword

Decsription: Lets you set a password for the wallet. In order to set this property, you must first create an obfuscated password for the wallet user account using the Apache server's `iasobf` command.

Valid values: obfuscated password for the wallet user

Scope: Instance

Example:

1. Obfuscate the password for the wallet user by entering the following on the Apache server:

   ```
   $ORACLE_HOME/Apache/Apache/bin/iasobf -p oracle1 root
   base64_encoded_string
   ```

   The result is a base64 encoded string, such as 03421BAB7EC4DFBD304495A4D56D541030CE4F7BB1DCA48C5.

2. Take the resulting obfuscated value and set the mxWalletPassword property:

   ```
   imtctl> setProperty -pname MxWalletPassword -pvalue string
   ```

SSLRequiredForMeetings

Description: Lets an administrator require HTTPS for all conferences or set HTTPS as the default option for all conferences.

Default value: false

Valid values: false/true

Scope: System, Site

Examples:

- To require that all conferences use HTTPS by default, but allow sites to override this setting, enter:

```
imtctl> setProperty -system true -pname SSLRequiredForMeetings -pvalue true
```

- To mandate use of HTTPS for all the conferences in a particular Oracle Web Conferencing site, enter:

```
imtctl> setProperty -siteId <site-id> -pname SSLRequiredForMeetings -pvalue
true -force true
```

User Controls to Set Secure Conferences

In the Oracle Web Conferencing Application, users can choose regular or SSL conferences (if both options are allowed by the administrator) by choosing the Regular or Restricted radio buttons in the **Schedule Details** tab. However, using controls provided in Oracle9*i*AS, an administrator can mandate use of SSL for some or all URLs in the application. Therefore, even if a user goes to a non-secure URL, the user could be redirected to a secure URL.

When setting their Web Conferencing Preferences, a user can mandate use of the secure mode for all attendees in the conferences that the user creates. Thus, attendees are forced to use the secure mode, based on the host's preferences.

Cobrowsing SSL Web sites

Users can use Oracle Web Conferencing to cobrowse SSL Web sites. No additional Web Conferencing configuration is necessary if the Web sites have Web server certificates from the following certificate authorities: RSA, GTE CyberTrust, Baltimore Technologies, and Entrust.

This set is not comprehensive. If you cobrowse a Web site that uses a certificate from a certificate authority not included in the default set, the Web Conferencing Console displays an error message such as:

```
IMT_SRV_CM_SSL_HANDSHAKE_FAILED.
The certificate submitted by the site has either expired or is not trusted by the
iMeeting server.Please request your iMeeting server administrator to update the
Trusted Signer Certificates Database with the certificate for the site you are
visiting.
[https://www.secureWeb site.com/uri/]
...
```

If you get this error, add Trusted Root Certificate Authority certificates to the Real-Time Collaboration Core Components machine as described in the next section.

Adding Trusted Root Certificate Authority Certificates

SSL communications involve, among other things, digital certificates. Certificates are issued by a third party, called a Certificate Authority (CA). When you use your browser to visit an SSL-secured Web site, your browser validates that the Web site is who it claims to be by verifying the Web site's certificate. Your browser comes bundled with a set of Trusted Root Certificate Authority Certificates, which it uses to validate any certificate a Web site presents. Occasionally, your browser will prompt you to determine whether you want to proceed, and also optionally installs a certificate when a Web site presents a new certificate signed by a CA.

The Web Conferencing Server that browses Web sites on behalf of conference attendees comes with a default set of Trusted Root Certificate Authority certificates. When an

attendee cobrowses a Web site that deploys a certificate from a CA vendor not included in the default set, you need to add the Trusted Root Certificate to the Real-Time Collaboration Core Components configuration. Follow the steps in this section to do this.

You will need to use Internet Explorer 5.5 or later to install the CA root certificate into the browser and export it as a Base64 encoded X.509 (.CER) file. You cannot use Netscape, because it does not allow the export of the root certificates to a file.

1. From an Internet Explorer browser that is not using Oracle Web Conferencing, go to the SSL-secured Web site that gave an error through Web Conferencing.
2. Using your browser, access secured areas of the Web site until you see the yellow lock in your browser status bar at the bottom.
3. Click the yellow lock, and then click the **Certification Path** tab in the pop-up window.
4. Select all the certificates one by one from the certificate chain. Click the **General** tab to read the description of the selected certificate. It contains the name of the issuer, its validity period, and so on.
5. If you are comfortable trusting the issuer (CA), then click the **Details** tab.
6. Click **Copy To File** and then click **Next** on the following screen.
7. In the subsequent screen, choose the format you want to export in as Base64 encoded X.509 (.CER) and click **Next**.
8. Enter a file name in which you want to store the certificate, for example, cavendor.cer, and save the CA certificate. Repeat steps 4 to 8 for all the certificates in the chain.

   Each of the files contains a certificate in the format:

   ```
   -----BEGIN CERTIFICATE-----
   MIICYzCCAdACEAuZ4ibKgW0O66lustIp2TowDQYJKoZIhvcNAQEEBQAwXzELMAkG
   A1UEBhMCVVMxIDAeBgNVBAoTF1JTQSBEYXRhIFNlY3VyaXR5LCBJbmMuMS4wLAYD
   VQQLEyVTZWN1cmUgU2VydmVyIENlcnRpZmljYXRpb24gQXV0aG9yaXR5MB4XDTAw
   ...
   -----END CERTIFICATE-----
   ```

9. Append the file(s) saved in step 8 to the `certdb.txt` file specified by the Real-Time Collaboration Core Components default property `TrustDBFile` (use the `imtctl` command `getProperty` to see the value of this property). This file is typically located in `$ORACLE_HOME/imeeting/conf`.

Properties to Configure Clusters

As discussed in "Real-Time Collaboration Clusters" on page 3-5, you can create clusters of Real-Time Collaboration core components. You do so by assigning separate `InstanceLocation` properties to a set of components. For example, if components A and B have an `InstanceLocation` of "US," and components C and D have an `InstanceLocation` of "India," you have two clusters, one labeled US and one labeled India.

Instances containing the Voice Conversion Server and Document Conversion Server can also be clustered by assigning them an InstanceLocation that matches the location of the Real-Time Collaboration core components you want them to serve.

Although not recommended, if for any reason an instance containing a Voice or Document Conversion server needs to serve multiple locations, use the `AdditionalLocationsServed` property in this chapter.

InstanceLocation

Description: Sets the location of an instance. By default, all instances are in the location `default`. All core services (Web Conferencing Servers) in a particular location are considered for load balancing.

Default value: Default

Valid values: Any string

Scope: Instance

Example: To set the location for an instance to "US," enter:

```
imtctl> setProperty -pname InstanceLocation -pvalue "US"
```

AdditionalLocationsServed

Description: Sets a list of locations to which a Voice or Document Conversion Server instances will provide service.

Default value: default

Valid values: The value should be in the format: "[\"location1\", \"location2\"]"

Scope: Instance

Example: To have a Document and Voice Conversion Server instance provide service to instances with locations HQ and UK, enter:

```
imtctl> setProperty -pname AdditionalLocationsServed -pvalue "[\"HQ\", \"UK\"]"
```

Properties to Configure E-mail Invites

The Oracle Web Conferencing Application can generate outbound e-mail for various purposes, such as inviting users to a conference. To enable this feature, you must configure the Real-Time Collaboration system with certain e-mail-related properties. See "Setting Up Web Conferencing" on page 4-3 for details about the steps to set up e-mail, including using an `imtctl` script to set the following properties.

SmtpHost

Description: The host name of the SMTP server Oracle Web Conferencing should use to send e-mail.

Default value: None

Valid values: a host name

Scope: System, Instance

Examples:

- To set the SMTP host for the whole system to mail-net.company.com, enter:

  ```
  imtctl> setProperty -system true -pname SmtpHost -pvalue "mail-net.company.com"
  ```

- To set the SMTP host for the current instance to mail-net.company.com, enter:

  ```
  imtctl> setProperty -pname SmtpHost -pvalue "mail-net.company.com"
  ```

SmtpPort

Description: The listen port of the SMTP server.

Default value: 25

Scope: System, Instance

Examples:

- To set SmtpPort for the whole system to 3000, enter:

```
imtctl> setProperty -system true -pname SmtpPort -pvalue 3000
```

- To set SmtpPort for the current instance to 3000, enter:

```
imtctl> setProperty -pname SmtpPort -pvalue 3000
```

EmailEnabled

Description: Enables the sending of e-mailed conference invitations.

Default value: false

Valid values: false, true

Scope: System, Instance

Example: To enable e-mailed conference invitations for an entire system, enter:

```
imtctl> setProperty -system true -pname EmailEnabled -pvalue true
```

AdminEmail

Description: Sets the e-mail account that should receive administrative e-mails from the Web Conferencing system

Default value: None

Valid values: Any user's e-mail address

Scope: System, Instance

Example: To set Joe Smith to receive administrative e-mails, enter:

```
imtctl> setProperty -system true -pname AdminEmail -pvalue joe.smith@oracle.com
```

Properties to Configure the Application Pages

You can customize various elements of the Oracle Web Conferencing Application. These customizations can have a system scope and also a site scope: a site can supersede the customizations of the system.

Customizing Login Messages and Public Meeting Tables

PreLoginMessage

Description: Sets the text of a message that appears under the login link on the login page.

Default value: "Login is based on Single Sign-On. To log in, enter your global e-mail ID (first.last@oracle.com) and your e-mail apssword."

Valid values: Any text message.

Scope: System, Site

Example: To set the login link text to "Log in to OracleWeb Conferencing" enter:

```
imtctl> setProperty -system true -pname PreLoginMessage -pvalue "Log in to Oracle
Web Conferencing"
```

PublicMeetingNumRows

Description: Sets the number of rows that are displayed in the Public Meetings table.

Default value: 10

Valid values: -1 to display all meetings, or any positive integer.

Scope: System, Site

Example: To show all meetings in the Public Meetings table for a particular site (siteID 123456), enter:

```
imtctl> setProperty -siteID 123456 -pname PublicMeetingNumRows -pvalue -1
```

PublicMeetingSortBy

Description: Sets the "sort by" column of the Public Meeting table.

Default value: The number of the "Start Date" column.

Valid values: Any integer indicating the column position, starting from 0.

Scope: System, Site

Example: To sort the Public Meetings table for a whole Web Conferencing system by the second column, enter:

```
imtctl> setProperty -system true -pname PublicMeetingSortBy -pvalue 1
```

PublicMeetingSortOrder

Description: Sets the sort order of the column specified in the `PreLoginPublicMeetingSortBy` property.

Default value: asc

Valid values: asc=ascending, desc=descending

Scope: System, Site

Example: To sort the rows in descending order in the `PublicMeetingSortBy` column, enter:

```
imtctl> setProperty -system true -pname PublicSortOrder -pvalue desc
```

Customizing the Privacy Statement

A default privacy statement ships with Oracle Web Conferencing. A link to the statement labeled "Privacy Statement" appears at the bottom of Web Conferencing application pages. Your company should evaluate whether the statement should be changed for your own needs; if so, you can change it using the following properties.

ShowPrivacyLink

Description: Sets whether the privacy link appears at the bottom of the Web Conferencing application pages.

Default value: true

Valid values: false, true

Scope: System, Site

Example: To suppress the privacy link for site ID 123456, enter:

```
imtctl> setProperty -siteID 123456 -pname ShowPrivacyLink -pvalue false
```

PrivacyLink

Description: Sets the URL to which the privacy link points.

Default value: cmn_privacy.uix

Valid values: Any valid URL, either a relative link from the web conferencing home or an absolute link in the format http://www.site.com/filename.html

Scope: System, Site

Example: To set the privacy link to go to the Web page www.mycompany.com/legal/privacy.html, enter:

```
imtctl> setProperty -system true -pname PrivacyLink -pvalue
"www.mycompany.com/legal/privacy.html"
```

PrivacyText

Description: Sets the text displayed in the privacy statement.

Default value: ORACLE RECOMMENDS EACH COMPANY DISCUSS THE LEGAL ISSUES OF USING ORACLE WEB CONFERENCING WITH ITS CUSTOMERS, PARTNERS, AND EMPLOYEES AND REPLACE THE TEXT ON THIS PAGE WITH THE APPROPRIATE DISCLAIMER. FOR INSTRUCTIONS ON HOW TO CHANGE THIS TEXT, PLEASE REFER TO THE ORACLE WEB CONFERENCING ADMINISTRATOR'S GUIDE.

Valid values: Any valid string.

Scope: System, Site

Example: To set the privacy text to "My Company Privacy text," enter:

```
imtctl> setProperty -system true -pname PrivacyText -pvalue "My Company Privacy
text"
```

Suppressing the Schedule Tab for Web Calendar Users

If your company uses the Oracle Collaboration Suite Calendar in conjunction with Web Conferencing, you may wish to require users to schedule conferences solely through the calendar. To do so, you set the following property to prevent the **Schedule** tab from appearing on the Web Conferencing application pages.

IsCalendarOCSInstalled

Description: Sets whether the **Schedule** tab appears.

Default value: false

Valid values: false, true

Scope: System

Example: To prevent the **Schedule** tab from appearing, so that users must schedule web conferences using the Oracle Calendar, enter the following:

```
imtctl> setProperty -system true -pname IsCalendarOCSInstalled -pvalue true
```

Configuring the Quicklinks Bin

You can make commonly-used URLs easily accessible by adding them to the Quicklinks bins on the Web Conferencing prelogin and home pages. You can add four user-defined links to this bin, setting the link names and URL with the following properties.

QuicklinkNName

Description: Sets the name of the Quicklink, where N is any number from 1 through 4.

Default value: Quicklink1, Quicklink2, Quicklink3, Quicklink4

Valid values: Any string.

Scope: System, Site

Example: To configure the first user-defined link to be called "Real-Time Meetings," enter:

```
imtctl> setProperty -system true -pname Quicklink1Name -pvalue "Real-Time
Meetings"
```

QuicklinkNURL

Description: Sets the URL associated with QuicklinkNName, where N is any number from 1 through 4.

Default value: None

Valid values: Any valid URL

Scope: System, Site

Example: To configure the first user-defined link to jump to http://realtimemeetings.company.com, enter:

```
imtctl> setProperty -system true -pname Quicklink1URL -pvalue
"http://realtimemeetings.company.com"
```

Note: If the value associated with any of the Quicklinks is null, then that link does not display in the Quicklinks bin.

Properties to Configure Conference Runtime Characteristics

You can configure some characteristics of a conference session using the following properties. These properties are used when a conference is created from an integrating site using the Oracle Web Conferencing Integration Service.

Note that individual users can also set many of these items for their own conferences, using the Preferences page.

StartupMeetingMode

Description: Sets the default startup mode of the Web Conferencing Console when it loads.

Default value: Cobrowse

Valid values: Cobrowse, DocumentPresentation, Whiteboard, DesktopSharing

Scope: Site

Example: To set conferences at site 1010506 to always start in DesktopSharing mode, enter:

```
imtctl> setProperty -siteId 1010506 -pname StartupMeetingMode -pvalue
DesktopSharing
```

EnableDesktopSharingMode

Description: Allows or prevents desktop sharing in conferences.

Default value: true

Valid values: true, false

Scope: System, Site

Example: To prevent Desktop Sharing mode for conferences at site 1010506, enter:

```
imtctl> setProperty -siteId 1010506 -pname EnableDesktopSharingMode -pvalue false
```

EnableWhiteboardMode

Description: Allows or prevents using Whiteboard mode in conferences.

Default value: true

Valid values: true, false

Scope: System, Site

Example: To allow Whiteboard mode for conferences at site 1010506, enter:

```
imtctl> setProperty -siteId 1010506 -pname EnableWhiteboardMode -pvalue true
```

EnableDocumentPresentationMode

Description: Allows or prevents document presentation in conferences.

Default value: true

Valid values: true, false

Scope: System, Site

Example: To prevent Document Presentation mode for conferences at site 1010506, enter:

```
imtctl> setProperty -siteId 1010506 -pname EnableDocumentPresentationMode -pvalue
false
```

EnableCobrowseMode

Description: Allows or prevents users from cobrowsing websites in conferences.

Default value: true

Valid values: true, false

Scope: System, Site

Example: To allow cobrowsing of other websites for conferences at site 1010506, enter:

```
imtctl> setProperty -siteId 1010506 -pname EnableCobrowseMode -pvalue true
```

MeetingAutoHide

Description: Sets whether the conference console should be automatically hidden at start-up.

Default value: false

Valid values: true, false

Scope: Site

Example:

```
imtctl> setProperty -siteId 1010506 -pname MeetingAutoHide -pvalue true
```

> **Note:** Be careful when setting this property. If users do not see the conference console, they may not know that they have successfully entered a web conference.

MeetingExpandHelpText

Description: Allows or prevents users from expanding the help text in the conference console.

Default value: false

Valid values: false, true

Scope: Site

Example: To allow expanded help text to appear in the conference console for meetings at site 1010506, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingExpandHelpText -pvalue true
```

MeetingExpandSecondRow

Description: Allows or prevents users from expanding the second row of icons and controls in the conference console. The second row includes the Network connection readout, security icon, and the Poll, Preferences, and Conference Details buttons.

Default value: false

Valid values: false, true

Scope: Site

Example: To allow the second row of icons to appear in the conference console for meetings at site 1010506, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingExpandSecondRow -pvalue true
```

MeetingAllUseTools

Description: Allows or prevents all conference attendees from using annotation tools.

Default value: true

Valid values: false, true

Scope: Site

Example: To prevent users participating in conferences at site 1010506 from using the annotation tools, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingAllUseTools -pvalue false
```

MeetingAllControlLayout

Description: Allows or prevents all attendees from controlling the console layout.

Default value: true

Valid values: false, true

Scope: Site

Example: To allow all attendees to control console layout in conferences at site 1010506, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingAllControlLayout -pvalue true
```

MeetingAllViewAttendeeList

Description: Allows or prevents attendees from viewing the attendee list.

Default value: true

Valid values: false, true

Scope: Site

Example: To prevent users at site 1010506 from viewing the attendee list while participating in conferences, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingAllViewAttendeeList -pvalue
false
```

MeetingAllViewStatusMsg

Description: Allows or prevents attendees from viewing status messages.

Default value: true

Valid values: false, true

Scope: Site

Example: To prevent users at site 1010506 from viewing conference status messages, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingAllViewStatusMsg -pvalue false
```

MeetingChatType

Description: Sets the default chat type for a conference

Default value: PUBLIC_HOST (public chat with the host)

Valid values: ALL (group chat among all attendees), NONE (no chat allowed), HOST (private chat with the host), PUBLIC_HOST (public chat with the host)

Scope: Site

Example: To set the default chat mode for conferences at site 1010506 to allow all attendees to participate in a group chat, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingChatType -pvalue ALL
```

MeetingChatInterface

Description: Sets whether the chat window is docked to the console, or a pop-up window.

Default value: P (pop-up)

Valid values: P, D (docked)

Scope: Site

Example: To set the chat window to be docked in the console by default, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingChatInterface -pvalue D
```

EnableChat

Description: Allows or prevents the Chat feature from being available during a conference

Default value: true

Valid values: true, false

Scope: System, Site

Example: To prevent users at site 1010506 from using the Chat feature during a conference, enter:

```
imtctl> setProperty -siteId 1010506 -pname EnableChat -pvalue false
```

EnableRecording

Description: Allows or prevents the recording features from being available during a conference

Default value: true

Valid values: true, false

Scope: System, Site

Example: To prevent attendees from recording a conference, enter:

```
imtctl> setProperty -system true -pname EnableRecording -pvalue false
```

EnableSharedControl

Description: Allows or prevents shared control between host and a selected attendee

Default value: true

Valid values: true, false

Scope: System, Site

Example: To prevent a host from sharing control with attendees in conferences at site 1010506, enter:

```
imtctl> setProperty -siteId 1010506 -pname EnableSharedControl -pvalue false
```

MeetingSharedControlUseTools

Description: Allows attendees with shared control to use annotation tools.

Default value: true

Valid values: true, false

Scope: Site

Example: To allow attendees of conferences at site 1010506 who have been granted shared control of the conference to use annotation tools, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingSharedControlUseTools -pvalue
true
```

MeetingSharedControlSendPoll

Description: Allows attendees with shared control to send polls.

Default value: false

Valid values: true, false

Scope: Site

Example: To allow attendees of conferences at site 1010506 who have been granted shared control of the conference to send polls, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingSharedControlSendPoll -pvalue
true
```

MeetingStartUrl

Description: Sets a default URL that is displayed when the conference starts.

Default value: none

Valid values: Any valid URL

Scope: Site

Example: To set the starting URL for meetings at site 1010506 to www.mycompany.com, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingStartUrl -pvalue
http://www.mycompany.com
```

MeetingEndUrl

Description: Sets a default URL that is displayed when the conference finishes.

Default value: none

Valid values: Any valid URL

Scope: Site

Example: To set the ending URL for meetings at site 1010506 to www.mycompany.com, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingEndUrl -pvalue
http://www.mycompany.com
```

MeetingIsInstant

Description: Sets whether conferences held at a site are instant conference.s

Default value: false

Valid values: true, false

Scope: Site

Example: To set conferences at site 1010506 to be standard rather than instant conferences, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingIsInstant -pvalue false
```

MeetingDurationMinutes

Description: Sets the suggested default conference duration in minutes.

Default value: 60

Valid values: any valid integer

Scope: Site

Example: To set scheduled conferences at site 1010506 to default to 30 minutes long, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingDurationMinutes -pvalue 30
```

MeetingTypeOfMtg

Description: Set the default conference type option in the **Schedule** tab.

Default value: REGULAR

Valid values: REGULAR (anyone to whom you provide the conference URL may attend), PUBLIC (anyone may attend, and the conference is published on the Web Conferencing pages), RESTRICTED (only registered users who have been invited may attend the conference; the conference is not listed on the Web Conferencing pages)

Scope: Site

Example: To set the default conference type at site 1010506 to Public, enter:

```
imtctl> setProperty -siteId 101 -pname MeetingTypeOfMtg -pvalue PUBLIC
```

MeetingDialInAndConfId

Description: Sets default dial-in information and a conference ID; this can be used if you have a phone conference service that you use while you web conference.

Default value: none

Valid values: Any text

Scope: Site

Example: To set the default dial-in for site 1010506 to dial a conference number and a specific conference id, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingDialInAndConfId -pvalue
"18001234567,,,ID555666#"
```

MeetingInformation

Description: Sets default additional information related to all conferences.

Default value: none

Valid values: Any text

Scope: Site

Example: To inform users at site 1010506 that SSL security is on by default, you could enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingInformation -pvalue "All
conferences at ABC Corp. use SSL security."
```

MeetingPublishOption

Description: Allows or prevents the host from publishing the attendee list.

Default value: true

Valid values: true, false

Scope: Site

Example: To prevent any hosts at site 1010506 from publishing attendee lists for their meetings, enter:

```
imtctl> setProperty -siteId 1010506 -pname MeetingPublishOption -pvalue false
```

MeetingEmailJoin

Description: Allows attendees to join the conference directly from the e-mail link.

Default value: true

Valid values: true, false

Scope: Site

Example: To allow attendees of conferences at site 1010506 to join conferences from the link displayed in an e-mail announcement, enter:

```
imtctl> setProperty -siteId 101 -pname MeetingEmailJoin -pvalue true
```

MeetingShouldSendEmail

Description: Allows the host to notify attendees by e-mail.

Default value: true

Valid values: true, false

Scope: Site

Example: To allow hosts of conferences at site 1010506 to send e-mailed invitations to a web conference, enter:

```
imtctl> setProperty -siteId 101 -pname MeetingShouldSendEmail -pvalue true
```

EarlyJoinMinutes

Description: Sets the number of minutes before a conference's scheduled start time during which users can join.

Default value: 30

Valid values: Any positive integer

Scope: System

Example: To configure this value to 15 minutes, enter:

```
imtctl> setProperty -system true -pname EarlyJoinMinutes -pvalue 15
```

AllowPublicMeetings

Description: Allows or prevents users from choosing Public meetings in the **Schedule** tab or listing archived conferences on the public Web page.

Default value: true

Valid values: false, true

Scope: System, Site

Example: To prevent users at site 1010506 from holding Public meetings or publishing archived meetings on the public web page, enter:

```
imtctl> setProperty -system true -pname AllowPublicMeetings -pvalue false
```

FirstAttendeeBehavior

Description: Sets whether the first attendee is prompted to be a presenter.

Default value: Attendee

Valid Values: Attendee (the attendee is not a presenter), DesktopControl (the attendee can present)

Scope: System, Site

Example: To set the first attendee to be prompted for presenter mode, enter:

```
imtctl> setProperty -system true -pname FirstAttendeeBehavior -pvalue
DesktopControl
```

Property to Configure Voice Conversion Servers

The `VoiceDialInPrefix` property lets you set dial-in prefixes. See "Setting Up Document and Voice Conversion Servers for Web Conferencing" on page 4-9 for complete details about all Voice Conversion server configuration.

VoiceDialInPrefix

Description: This property sets the dial-in prefix for an instance of the Voice Conversion Server. This prefix would be added to all dial-in numbers received by the Voice Conversion Server before dialing out. Typically, corporations have a dial-in prefix ("9," for example) for dialing numbers outside the company's internal phone network.

Default Value: None

Valid Value: Any valid dialing sequence. In most cases, it would be a number example: "9" A valid dialing sequence contains numbers 0-9 and any of the following characters: * # ,

Scope: Instance

Example: To configure the dial-in prefix for a Voice Conversion Server instance to "9" enter:

```
imtctl> setProperty -i <instance-name> -pname "VoiceDialinPrefix" -pvalue "9"
```

Property to Configure E-Mailed Usage Reports

Oracle Web Conferencing reports can be generated and sent via e-mail using the imtreport script. This script also controls whether the Usage Reports that appear under the **Reports** tab are populated with data. "Setting Up Web Conferencing" on page 4-3 describes how to edit the script and set up a cron job or Scheduled Task to run the script on a regular basis.

The ReportEnvironmentName property controls the name that appears at the top of the e-mailed reports.

ReportEnvironmentName

Description: Sets the name of the environment for this report. The name appears in the report header.

Default value: None

Valid values: Any text

Scope: System

Example: To set the report environment name to "My Company Web Conferencing System," enter:

```
imtctl> setProperty -system true -pname ReportEnvironmentName -pvalue "My Company
```

```
Web Conferencing System"
```

Properties to Configure Logging

You can configure levels of conference logging, as well as the location of the generated logs.

LogLevel

Description: Log level

Default value: CONFIG

Valid values: NONE, SEVERE, WARNING, INFO, CONFIG, FINE, FINER, FINEST, ALL

Scope: System, Instance and Component. The property can be set at a system scope, can be overridden for a specific instance and in turn can be overridden for a specific component in that instance.

Examples:

An administrator wants to set the system-wide log level to SEVERE and then for a specific instance, the administrator wants to change it to WARNING and then for the Web Conferencing Servers in that instance, wants to change the log level to FINE.

- To set the system-wide log-level to SEVERE, enter:

  ```
  imtctl> setProperty -system true -pname LogLevel -pvalue SEVERE
  ```

- To set the instance-wide log-level for the current instance to WARNING, enter:

  ```
  imtctl> setProperty -pname LogLevel -pvalue WARNING
  ```

- To set the log level for Web Conferencing Servers only to FINE in this instance, enter:

  ```
  imtctl> setProperty -ct clbsvr -pname LogLevel -pvalue FINE
  ```

LogPath

Description: The log path relative to IMT_HOME. The filename must be specified using forward slashes '/', even on the Microsoft Windows platform.

Default value: logs (Because this is relative to IMT_HOME, it becomes IMT_HOME/logs)

Valid values: any directory. The directory name must be specified using forward slashes (/), even on the Microsoft Windows platform.

Scope: Component (default), System, Instance

Example: To set the log path to directory IMT_HOME/imtlogfiles/, enter:

```
imtctl> setProperty -system true -pname LogPath -pvalue "imtlogfiles"
```

LogFlushInterval

Description: System-wide log flush interval in the system-wide log flush interval, duration types must be specified in some combination of days, hours, minutes, seconds, and milliseconds. Valid abbreviations include "d", "hr", "min", "sec", "millis". Values must be in quotation marks due to the spaces.

Default value: 10 sec

Scope: Component (default), System, Instance

Example: To set the log flush interval to one second, enter:

```
imtctl> setProperty -pname LogFlushInterval -pvalue "1 sec"
```

Property to Configure Time Zones

Users can set their time zone by logging in to the Web Conferencing Application and clicking Preferences. A list of available time zones displays.

After users have set their time zone, all date and time references in the Web Conferencing Application and Web Conferencing Console are automatically converted to their time zone.

Example: User A sets his time zone to "(-05:00) US Eastern Time." He then schedules a conference for 30-Mar-2004 5:00 PM. He intends to have the conference at 5 pm Eastern Time. User B, whose time zone is set to "(-08:00) US Pacific Time," is invited to this conference. When User B sees the invitation in her Upcoming Meetings tab, it displays as 30-Mar-2004 2:00 PM. The time has been converted to Pacific Time for User B.

See Appendix A, "Time Zones" for a complete list of supported time zones.

When a user registers with Oracle Web Conferencing for the first time, the user's default time zone is set based on the system-level property DefaultTimeZoneName.

DefaultTimeZoneName

Description: Sets the default time zone for this Web Conferencing system.

Default: UMT (Universal Mean Time)

Valid values: Any zone listed in Appendix A, "Time Zones"

Scope: System

Example: To set the system time zone to New York time, enter:

```
imtctl> setProperty -system true -pname DefaultTimeZoneName -pvalue America/New_
York
```

Note: This setting only affects the default time zones for users who have not set their time zone yet. You should encourage users to set their time zones.

6

Sample Deployments

This chapter discusses some sample deployment scenarios, ranging from a basic deployment to a sophisticated one spanning multiple geographical regions. It also explains what an administrator should do to set them up. The best way to review this chapter is to go through each example, because each example builds on the previous.

> **Note:** In Chapter 4, "Post-Installation" Oracle Web Conferencing properties are set using .imt files. You can also set properties using the `setProperty` command in `imtctl`, as shown in some examples in this chapter. It is important to remember that when you use the option `-system true`, the setting of the property has a global scope and affects not just the instance on which `imtctl` is being invoked. However, if the `-system true` option is not used in the command, the command applies only to that instance.

See Chapter 4, "Post-Installation" for basic information about configuration Oracle Web Conferencing.

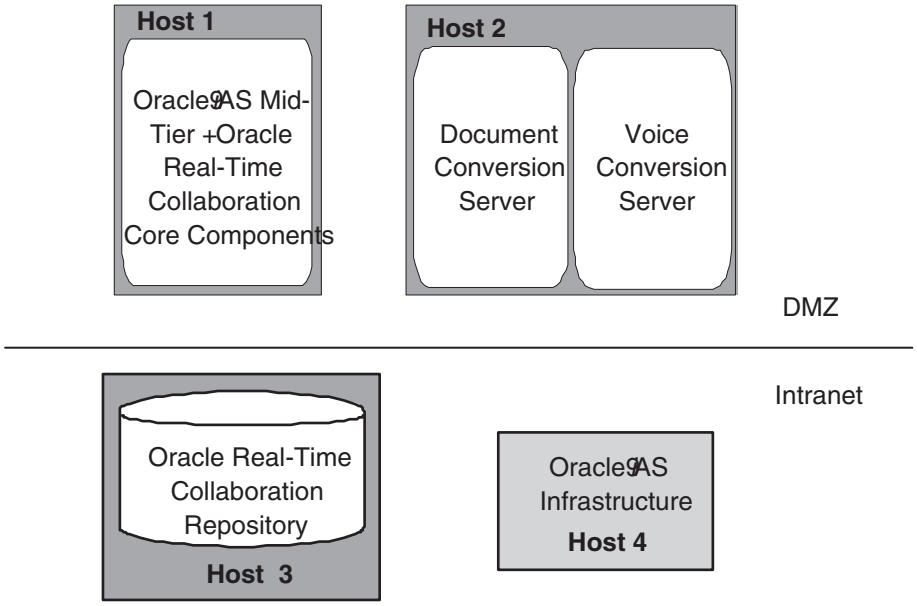
For all examples in this chapter: I

- One Oracle9*i*AS instance is on the machine with the Real-Time Collaboration Core Components and only one Real-Time Collaboration instance is on this machine.
- It is assumed that Oracle9iAS Web Cache is disabled on the middle tier.

Basic Deployment

This example shows the simplest deployment and is generally used to create a pilot deployment. In this deployment, users can access Oracle Web Conferencing from the Internet, through their proxy across the Internet, and through the intranet.

Figure 6–1 Web Conferencing Basic Deployment



In Figure 6–1, "Web Conferencing Basic Deployment", Host 1 and Host 2 are deployed in the same DMZ, while Host 3 and Host 4 are deployed in the corporate intranet.

These are the mandatory associations that must be made for the various parts of Oracle Web Conferencing to work:

- Real-Time Collaboration Core Components to Real-Time Collaboration Repository. This association is made during installation.
- Real-Time Collaboration Core Components to Oracle9*i*AS Infrastructure. This association is made during installation.
- Document and Voice Conversion Servers to Real-Time Collaboration Repository. This association is made during installation.

> **Note:** For these examples, the Voice Conversion Server and the Document Conversion Server are on the same machine.

- Document and Voice Conversion Servers to Real-Time Collaboration Core Components. This is a mandatory post-installation step.

Additional configuration issues are discussed in the rest of the chapter.

Sequence of Steps

1. Install all prerequisites on all the computers.
2. Install the Real-Time Collaboration Core Components on Host 1.
3. Perform post-installation steps on Host 1.
4. Perform verification tests on Host 1.
5. Install Document and Voice Conversion Servers on Host 2.
6. Follow the post-installation steps on Host 2.

7. Perform verification tests on Host 2.
8. Perform verification tests on Host 1 to confirm proper connection to Host 2.

Prerequisites

Host 1: Oracle9*i*AS plus Real-Time Collaboration Core Components set up and configured to use the Oracle9*i*AS Infrastructure on Host 4.

Host 4: Oracle9*i*AS Infrastructure needs to be set up on this machine.

Host 2: A Windows NT computer with specialized telephony hardware and software.

Host 3: Oracle9*i* Database (9.2.0.1) for the Real-Time Collaboration Repository.

> **See Also:** Oracle Collaboration Suite Installation Guide for Solaris for additional information on the prerequisites

Settings

Configure Oracle Web Conferencing to leverage existing enterprise e-mail and proxy servers. Assume that the settings are as follows:

Table 6–1 System Settings

Setting Name	Value	Comments	Web Conferencing Property
Existing e-mail server host name (one for the enterprise)	"mail-net.company.com"	Post-installation step	SmtpHost
E-mail server listening port	25	Default	SmtpPort
Existing proxy server host name for HTTP (one for the enterprise)	"www-proxy.company.com"	Post-installation step	UserAgentProxyHost
Proxy server host name for HTTPS (one for the enterprise)	"www-proxy.company.com"	Post-installation step	UserAgentProxySSLHost
Proxy port for HTTP	80	Default	UserAgentProxyPort
Proxy port for HTTPS	443	Default	UserAgentProxySSLPort
Proxy domains to exclude	".company.com"	Post-installation step	UserAgentProxyExclusions

Table 6–2 Instance on Host 1

Setting Name	Value	Comments	Web Conferencing Property
Number of mx	1	Default	SrvNumProcs
Listening port for the single mx	2400	Set during installation	MxListenPort
Actual host name of Host 1	"host1.company.com"		N/A
Alias for Host 1	"imeeting1.company.com"	Post-installation step	ApacheWebHost
Instance name	"instance1.company.com"	Set automatically during installation	N/A
Oracle HTTP Server listening port for HTTP on Host 1	7777	Default. Set during installation.	ApacheWebPort
Oracle HTTP Server listening port for HTTPS on Host 1	443	Default	ApacheWebSecurePort

You must configure the settings in Table 6–1 and Table 6–2 every time you install Web Conferencing.

In addition, you must configure Host 2 as follows:

Table 6–3 Host 2 Settings

Setting Name	Value	Comments	Web Conferencing Property
HTTP listening port for the Voice Conversion Server	2460	Set during installation	
Dial-in prefix for voice conversion server instance	9	Post-installation step	VoiceDialinPrefix

Installation

On Host 1, install the Real-Time Collaboration Core Components, and provide the required port numbers. On Host 2, install the Document and Voice Conversion Servers, and provide the required port numbers.

> **See Also:** Oracle Collaboration Suite Installation and Configuration Guide for Solaris for details

Post-Installation

The commands in this section are based on the data in the Settings tables in "Settings" on page 6-3.

Step 1 Integrate Web Conferencing with existing e-mail.

```
imtctl> setProperty -system true -pname SmtpHost -pvalue "mail-net.company.com"
```

Because the port is 25 (which is default), there is no need to set the port.

Step 2 Integrate Web Conferencing to use existing proxy servers.

```
imtctl> setProperty -system true -pname UserAgentProxyHost -pvalue
"www-proxy.company.com"
imtctl> setProperty -system true -pname UserAgentProxySSLHost -pvalue
"www-proxy.company.com"
imtctl> setProperty -system true -pname UserAgentProxyExclusions -pvalue
"[\".company.com\"]"
```

> **Note:** Because the default proxy port for HTTP is 80 and HTTPS is 443, there is no need to set these ports.

Step 3 Integrate Web Conferencing with Oracle HTTP Server on this Real-Time Collaboration Core Components machine.

Enable the mod_imeeting.

Make the necessary configuration changes to the Oracle HTTP Server in which mod_imeeting is configured. mod_imeeting is included through the $ORACLE_HOME/imeeting/conf/mod_imeeting.conf file, which in turn is included in $ORACLE_HOME/Apache/Apache/conf/oracle_apache.conf file for Oracle HTTP Server.

Add the following include line for mod_imeeting.conf in oracle_apache.conf:

```
Include "$ORACLE_HOME/imeeting/conf/mod_imeeting.conf"
```

Replace $ORACLE_HOME with the actual Oracle home.

1. Update the Oracle HTTP Server configuration using DCM.

   ```
   $ORACLE_HOME/dcm/bin/dcmctl updateConfig -ct ohs -v
   ```

2. Restart Oracle HTTP Server through DCM.

   ```
   $ORACLE_HOME/dcm/bin/dcmctl restart -ct ohs -v
   ```

3. Set up mod_imeeting/mx redirect.

   For one mx process (installer default), invoke:

   ```
   imtctl> setProperty -ct mxcomm -pname MxRedirectPort -pvalue "[\"2420\"]"
   ```

4. Make Web Conferencing aware of basic Oracle HTTP Server settings by invoking:

   ```
   imtctl> setProperty -pname ApacheWebHost -pvalue "imeeting1.company.com"
   ```

   Because 80 and 443 are defaults, no additional steps are required for setting the ports.

   Note: imeeting1.company.com is the URL users will use to access Web conferences.

Step 4 Set up reports.

Web Conferencing reports can be generated and sent via e-mail using the imtreport script. The script is located at $ORACLE_HOME/imeeting/bin and is invoked as "imtreport" on all platforms. The actual script file is "imtreport" for UNIX and Linux and "imtreport.cmd" for Microsoft Windows. Mandatory report options such as the recipient list are set by editing variables at the top of the report script. The script contains comments on the options provided. Be sure to edit the proper script for the platform you are using. You may want to send out automated weekly reports by using some operating system provided mechanism for invoking the script on a weekly basis (for example, cron job on UNIX or the Task Scheduler on Windows).

Step 5 Set up for periodic monitoring.

Set up your monitoring infrastructure to periodically ping the following URLs:

- For conference service availability monitoring:

  http://imeeting1.company.com/imtapp/servlet/ImtTestServlet?mtgtest=true

- For Voice Conversion Server availability monitoring:

  http://imeeting1.company.com/imtapp/servlet/ImtTestServlet?voiceconvtest=true

- For Document Conversion Server availability monitoring:

  http://imeeting1.company.com/imtapp/servlet/ImtTestServlet?docconvtest=true

See "Real-Time Collaboration Interfaces" on page 7-5 for more information.

Host 2

No post-installation configuration is required.

Verification

Host 1

Invoke `imtctl> runTests`. This runs the full verification test.

Typically, all tests succeed. However, if this command is run before Host 2 is set up, the Voice Conversion Server test and the Document Conversion Server test will fail.

Run this test as described in "Sequence of Steps" on page 6-2.

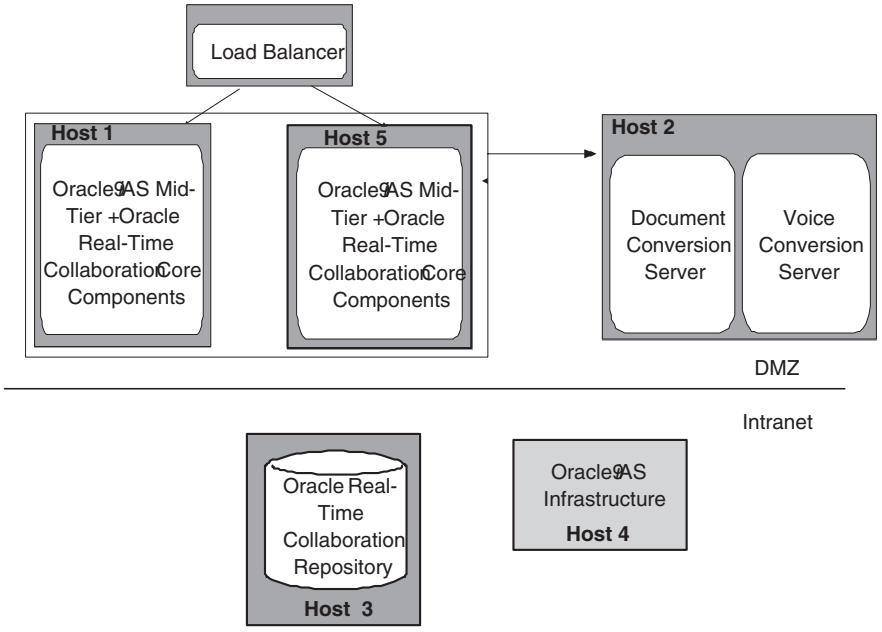
Host 2

Invoke `imtctl> getState` to verify that the processes are up, but the real test needs to be invoked on Host 1 (`imtctl> runTests`) to make sure that the document and voice conversion services provided by Host 2 are available.

Multiple Real-Time Collaboration Core Components and Load Balancer

In this deployment, Host 1, Host 2, and Host 5 are deployed in the DMZ, while Host 3 and Host 4 are deployed in the corporate intranet.

Figure 6–2 Deployment with Multiple Core Components and an LBR



Sequence of Steps

1. Install all prerequisites on all the computers, including the load balancer (LBR).
2. Install Real-Time Collaboration Core Components on Host 1.
3. Perform post-installation steps on Host 1.
4. Perform verification tests on Host 1.
5. Install Document Conversion Server and Voice Conversion Server on Host 2.
6. Perform post-installation steps on Host 2.

7. Perform verification tests on Host 2.
8. Perform verification tests on Host 1 to confirm proper connection to Host 2.
9. Install Real-Time Collaboration Core Components on Host 5.
10. Perform post-installation steps on Host 5.
11. Perform verification tests on Host 5.

Prerequisites

Host 1: Oracle9*i*AS mid-tier set up and configured to use the Oracle9*i*AS Infrastructure on Host 4

Host 5: Oracle9*i*AS mid-tier set up and configured to use the Oracle9*i*AS Infrastructure on Host 4

Host 4: Oracle9*i*AS Infrastructure needs to be set up on this machine.

Host 2: A Windows NT computer with specialized hardware and software

Host 3: Oracle9*i* Database for the Real-Time Collaboration Repository

Using the instructions in the Advanced Configuration document at http://otn.oracle.com/products/webconferencing/, configure the LBR/Oracle9iAS Single Sign-On for multiple sets of Real-Time Collaboration Core Components. The URL used to access Oracle Web Conferencing will now be "imeeting.company.com," and "imeeting1.company.com" and "imeeting2.company.com" will be behind the LBR. Users will access "imeeting.company.com" and the request will automatically be routed to either imeeting1.company.com or imeeting2.company.com.

> **See Also:** Oracle Collaboration Suite Installation and Configuration Guide for Solaris for additional information on the prerequisites

Settings

Configure Web Conferencing to leverage existing enterprise e-mail and proxy servers. Assume that the settings are as follows:

Table 6–4 System Settings

Setting Name	Value	Comments	Web Conferencing Property
Existing e-mail server host name (one for the enterprise)	"mail-net.company.com"	Post-installation step	SmtpHost
E-mail server listening port	25	Default	SmtpPort
Existing proxy server host name for HTTP (one for the enterprise)	"www-proxy.company.com"	Post-installation step	UserAgentProxyHost
Proxy server host name for HTTPS (one for the enterprise)	"www-proxy.company.com"	Post-installation step	UserAgentProxySSLHost
Proxy port for HTTP	80	Default	UserAgentProxyPort

Table 6–4 (Cont.) System Settings

Setting Name	Value	Comments	Web Conferencing Property
Proxy port for HTTPS	443	Default	UserAgentProxySSLPort
Proxy domains to exclude	".company.com"	Post-installation step	UserAgentProxyExclusions
Web Conferencing access URL	"imeeting.company.com"	Post-installation step. Requests to imeeting.company.com will be redirected to imeeting1.company.com and imeeting2.company.com.	GlobalWebHost

Host 1 settings are the same as described as in Table 6–2, " Instance on Host 1".

Host 5 settings are as follows:

Table 6–5 Host 5 Settings

Name of Setting	Value	Comments	Web Conferencing Property
Number of mx	1	Default	SrvNumProcs
Listening port for the single mx	2400	Set during installation. Keep it the same as Host 1.	MxListenPort
Actual host name of Host 5	"host5.company.com"		N/A
Alias for Host 5	"imeeting2.company.com"	Post-installation step	ApacheWebHost
Apache listening port for HTTP on Host 5	7777	Default. Set during installation.	ApacheWebPort
Apache listening port for HTTPS on Host 5	443	Default	ApacheWebSecurePort

Installation

On Host 1 and Host 5, install the Real-Time Collaboration Core Components, and provide the required port numbers.

Post-Installation

The commands in this section are based on the data in the Settings tables in "Settings" on page 6-7.

Host 1

Step 1 Set up GlobalWebHost.

```
imtctl> setProperty -system true -pname GlobalWebHost -pvalue
"imeeting.company.com"
```

Follow steps 2-7 as defined in "Post-Installation" on page 6-4.

Step 2 Set up e-mail.

Integrate Web Conferencing with e-mail, as before.

Step 3 Set up proxy.

Set up for existing proxy servers, as before.

Step 4 Integrate Web Conferencing with Oracle HTTP Server.

Integrate with the HTTP Server on this Real-Time Collaboration Core Components machine.

Step 5 Set up for monitoring.

Set up for monitoring, as before.

Step 6 Set up reports.

Set up Web Conferencing reports, as before.

Host 5

It is not necessary to set up e-mail, proxy, or global Web host, because they are global settings and will apply to this instance, as well.

Step 1 Integrate Web Conferencing with Oracle HTTP Server on this Real-Time Collaboration Core Components machine.

Follow the step as defined in "Post-Installation" on page 6-4.

Step 2 Set up for monitoring.

Follow the step as defined in "Post-Installation" on page 6-4.

Host 2

No post-installation steps are required.

Verification

Host 1

Invoke `imtctl> runTests`. This runs the full verification test.

Typically, all tests succeed. However, if you run this command before setting up Host 2, the Voice Conversion Server test and the Document Conversion Server test will fail.

Run this as described in the "Sequence of Steps" on page 6-6.

Host 5

Invoke `imtctl> runTests`. This runs the full verification test.

Typically, all tests succeed. However, if you run this command before setting up Host 2, the Voice Conversion Server test and the Document Conversion Server test will fail.

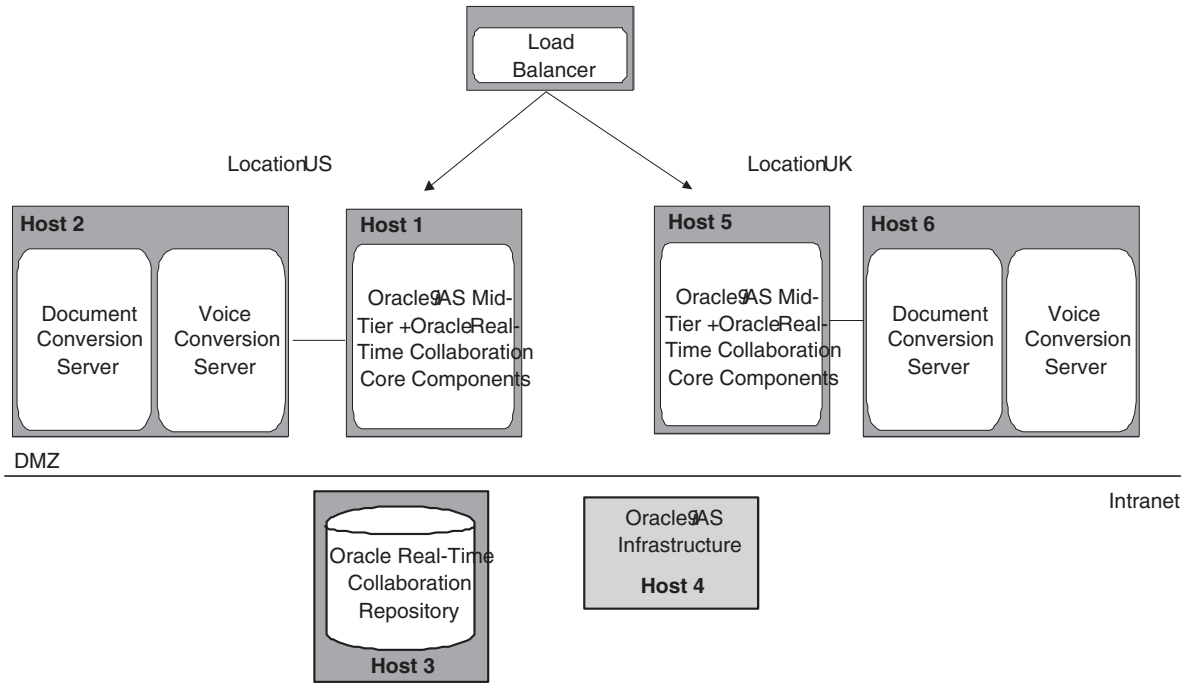
Run this as described in "Sequence of Steps" on page 6-6.

Host 2

Invoke `imtctl> getState` to verify that the processes are up, but the real test needs to be invoked on Host 1 and Host 5 (`imtctl> runTests`) to make sure that the services provided by Host 2 are available.

Multiple Geographical Locations and Load Balancer

Figure 6–3 Deployment with Multiple Geographical Locations and a LBR



In this deployment, Host 1, Host 2, Host 5, and Host 6 are deployed in the DMZ. Host 3 and Host 4 are deployed in the corporate intranet.

This is almost identical to the previous example ("Multiple Real-Time Collaboration Core Components and Load Balancer" on page 6-6) with the following key differences:

- LBR configuration: Host 1 and Host 5 are in different geographical locations, the United States and the United Kingdom. Furthermore, the LBR is configured to direct URL requests from users in the UK to the Real-Time Collaboration Core Components in the UK (Host 5), and URL requests from users in the US to the Real-Time Collaboration Core Components in the US. When users in the UK access imeeting.company.com, requests are redirected to imeeting1.company.com on Host 5. Similarly, when users in the US access imeeting.company.com, requests are redirected to imeeting2.company.com.
- Separate sets of Document and Voice Conversion Servers: The Real-Time Collaboration Core Components in Host 1 can use the Document and Voice Conversion Servers in Host 2 (location US) only, and the Real-Time Collaboration Core Components in Host 5 can use the Document and Voice Conversion Servers in Host 6 only.

Refer to Table 6–5 and Table 6–5 in the previous section for all the values for Host 1 through Host 5 and system settings. The changes are as follows:

Table 6–6 Host 5 Setting Changes from the Previous Section

Setting Name	Value	Comments	Web Conferencing Property
Location of Host 5	"UK"	Post-installation step.	InstanceLocation

Table 6–7 Host 6 Settings

Setting Name	Value	Comments	Web Conferencing Property
HTTP listening port for the Voice Conversion Server	2460	Set during installation.	
Location	"UK"	Post-installation step. Keep it the same as Host 5.	InstanceLocation

Table 6–8 Host 1 Setting Changes from the Previous Section

Setting Name	Value	Comments	Web Conferencing Property
Location	"US"	Post-installation step.	InstanceLocation

Table 6–9 Host 2 Settings

Setting Name	Value	Comments	Web Conferencing Property
HTTP listening port for the Voice Conversion Server	2460	Set during installation.	
Location	"US"	Post-installation step. Keep it the same as Host 1.	InstanceLocation

Follow all installation, prerequisite, and post-installation instructions from "Multiple Real-Time Collaboration Core Components and Load Balancer" on page 6-6 in addition to the following:

Host 1

Step 1 Set up the location.

```
imtctl> setProperty -pname InstanceLocation -pvalue "US"
```

Host 2

Set this up to serve the Real-Time Collaboration Core Components on Host 1.

```
imtctl> setProperty -pname InstanceLocation -pvalue "[\"US\"]"
```

Host 5

Step 1 Set up the location.

```
imtctl> setProperty -pname InstanceLocation -pvalue "UK"
```

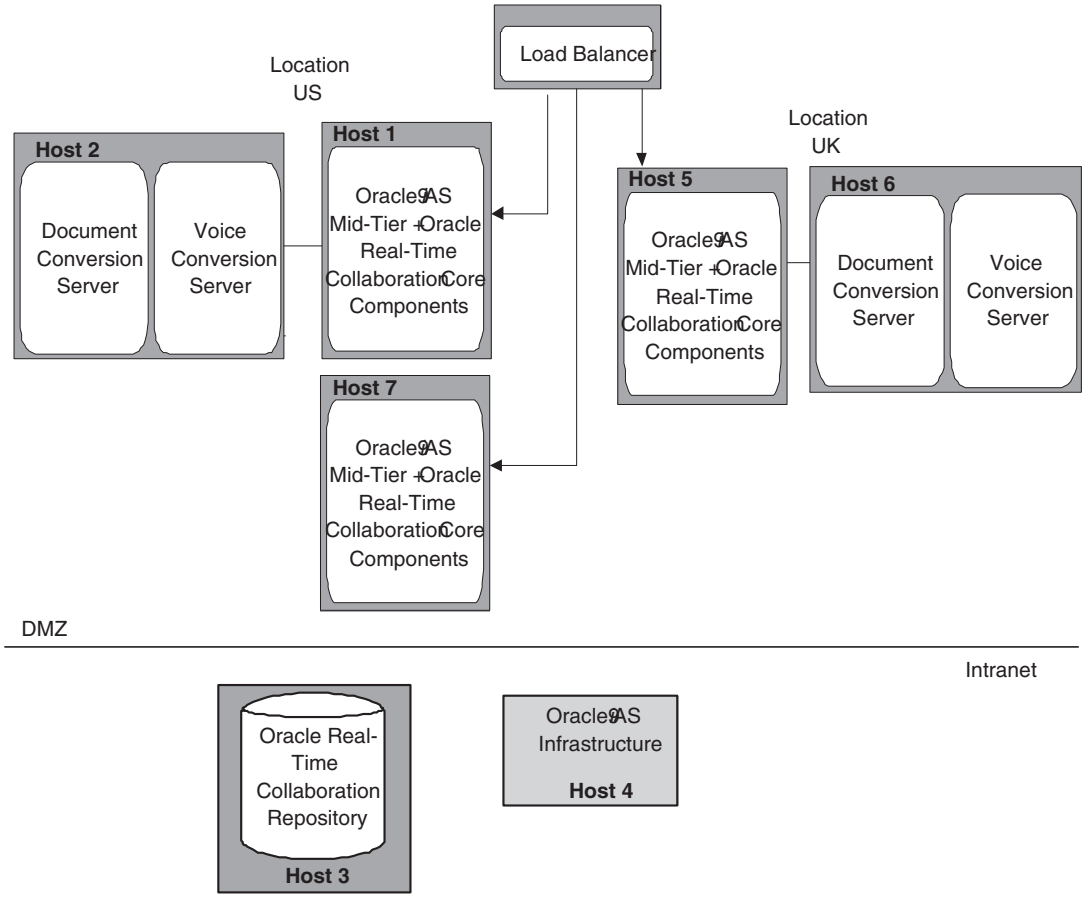
Host 6

Set this up to serve the Real-Time Collaboration Core Components on Host 5.

```
imtctl> setProperty -pname InstanceLocation -pvalue "[\"UK\"]"
```

Adding New Core Components to an Existing System

Figure 6–4 Existing Web Conferencing Deployment with Additional Core Components



This example shows the addition of a machine, Host 7, to the configuration shown in the previous example. The changes are as follows:

- Host 7 has been added in location "US"
- Host 2 will provide document and voice conversion services to the Real-Time Collaboration Core Components on Host 7.
- The LBR needs to be configured to direct all requests from the US to either Host 1 or Host 7 and to continue to direct all UK requests to Host 5.

Sequence of Steps

1. Install prerequisites on Host 7.
2. Install Real-Time Collaboration Core Components on Host 7.
3. Perform post-installation steps on Host 7.
4. Perform verification tests on Host 7.
5. Restart the Document and Voice Conversion Servers on Host 2.
6. Perform verification tests on Host 7.

Prerequisites

Host 7: Oracle9*i*AS mid-tier set up and configured to use the Oracle9*i*AS Infrastructure on Host 4.

> **See Also:** Oracle Collaboration Suite Installation and Configuration Guide for Solaris for details

Settings

Configure Host 7 as follows:

Table 6–10 Host 7 Settings

Name of setting	Value	Comments	Web Conferencing Property
Number of mx	1	Default	SrvNumProcs
Listening port for the single mx	2400	Set during installation. Keep it the same as Host 1.	MxListenPort
Actual host name of Host 7	"host7.company.com"		N/A
Alias for Host 5	"imeeting3.company.com"	Post-installation step	ApacheWebHost
Apache listen port for HTTP on Host 5	7777	Default. Set during installation.	ApacheWebPort
Apache listen port for HTTPS on Host 5	443	Default	ApacheWebSecurePort
Location of Host 5	"US"	Post-installation step. Keep it the same as Host 1.	InstanceLocation

Installation

On Host 1 and Host 5, install the Real-Time Collaboration Core Components, and provide the required port numbers.

Post-Installation

Invoke the imtctl commands in this section, while setting Web Conferencing properties. The commands in this section are based on the data in the Settings tables in "Settings" on page 6-13.

Host 7

It is not necessary to set up e-mail, proxy, or global Web host, because they are global settings and will apply to this instance, as well.

Step 1 Integrate Web Conferencing with Oracle HTTP Server on this Real-Time Collaboration Core Components machine.

Follow the step as defined in "Post-Installation" on page 6-4.

Step 2 Set up location.

```
imtctl> setProperty -pname InstanceLocation -pvalue "US"
```

Step 3 Set up for monitoring.

Follow the step as defined in "Post-Installation" on page 6-4.

Step 4 Restart Host 2.

Restart Host 2 so that it recognizes the Real-Time Collaboration Core Components on Host 7.

Verification

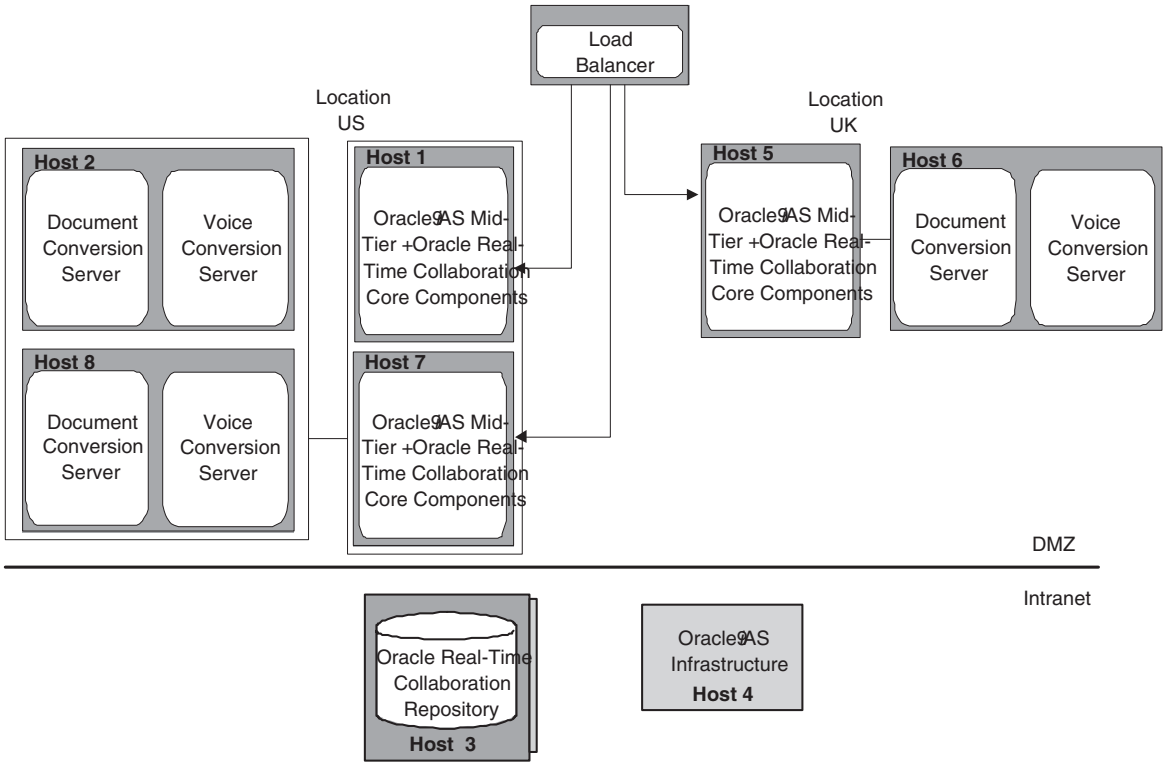
Host 7

Invoke`imtctl> runTests`. This runs the full verification test.

All tests succeed.

Adding Document and Voice Conversion Servers to an Existing System

Figure 6–5 Existing Deployment with Additional Document and Voice Conversion Servers



Another machine with Document and Voice Conversion Servers is added (Host 8).

Sequence of Steps

1. Install prerequisites on Host 8.
2. Install the Document and Voice Conversion Servers on Host 8.
3. Perform post-installation steps on Host 8.
4. Perform verification tests on Host 8.
5. Bring down the Document and Voice Conversion Servers on Host 2, so that Host 8 is the only one available.

6. Perform verification tests on Host 7 or Host 1.
7. Restart the Document and Voice Conversion Servers on Host 2.

Prerequisites

See the Oracle Collaboration Suite Installation and Configuration Guide for Solaris for additional information on the prerequisites.

Settings

Table 6–11 Host 8 Settings

Setting Name	Value	Comments	Web Conferencing Property
HTTP listening port for the Voice Conversion Server	2460	Set during installation	
Location	"US"	Post-installation step	InstanceLocation

Installation

On Host 8, install the Document and Voice Conversion Servers, and provide the required port numbers.

Post-Installation

Host 8

1. Set this up to serve the Real-Time Collaboration Core Components in location "US" by invoking:

```
imtctl> setProperty -pname InstanceLocation -pvalue "[\"US\"]"
```

1. Restart the Document and Voice Conversion Servers by invoking `imtctl> stop` and then `imtctl> start`.

Verification

Host 8

Invoke `imtctl> getState` to verify that the processes are up, but the real test needs to be invoked on Host 1 (`imtctl> runTests`) to make sure that the services provided by Host 8 are available.

Host 2

Shut down Host 2.

Host 1

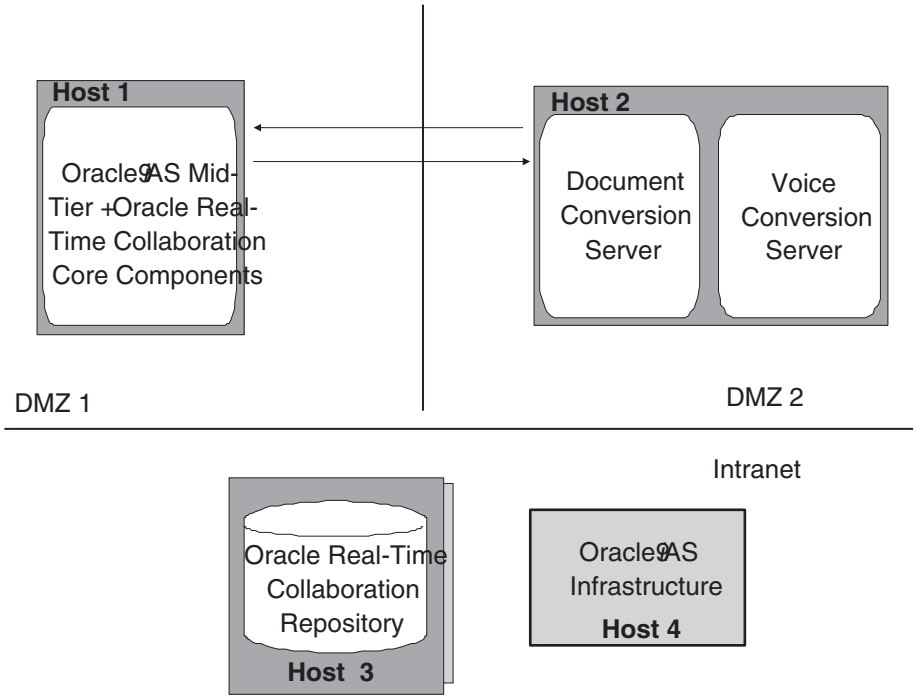
Invoke`imtctl> runTests`. This runs the full verification test.

All tests succeed.

Additional Example

The topology is identical to the basic deployment, but the Document and Voice Conversion Servers are deployed in a different DMZ.

Figure 6–6 Deployment with Core Components in a Different DMZ



The following additional configuration is required:

- From DMZ 1 to DMZ 2, enable HTTP requests on port 2450.
- From DMZ 2 to DMZ 1, enable TCP/IP requests on port 2400.

7

Monitoring

This chapter describes how you can monitor Real-Time Collaboration components to provide quality of service for conferences and uninterrupted availability for conference service. This chapter discusses:

- How the system monitors all Real-Time Collaboration processes and restarts them if they are down.
- How you monitor whether an instance in a system is capable of providing the direct services.
- How you monitor performance statistics for tuning the Real-Time Collaboration system
- How you monitor current conferences, including events for each conference.
- How to verify key configuration setup.
- Interfaces that allow Real-Time Collaboration services to be monitored by any monitoring system.

Process Monitoring

The Real-Time Collaboration Process Monitor (imt-pm) is a Java-based process that runs as a daemon, managing all Real-Time Collaboration processes in an instance. The Real-Time Collaboration Process Monitor opens an HTTP listening point to accept requests to start and stop processes. Specifically, it does the following:

- Periodically pings all Real-Time Collaboration processes in an instance, to check if the process is active.
- Automatically restarts inactive processes (processes that do not respond to the ping) without manual intervention.

The Oracle Process Management and Notification system monitors imt-pm and serves as a watchdog for it. The Oracle Process Management and Notification system automatically restarts imt-pm if it detects it to be inactive. If imt-pm goes down, there is no impact on the processes that were being monitored by imt-pm. imt-pm can recover its state without affecting the processes that it was monitoring before it went down.

imt-pm and start/stop

```
imtctl> start
```

When you invoke the start command with no arguments from imtctl, Oracle Process Management and Notification, and imt-pm (if necessary) automatically start, which then starts all the processes it needs to manage.

Other start commands require that imt-pm already be running.

```
imtctl >stop
```

The stop command with no arguments (stopall behavior) shuts down imt-pm and all of the processes it is managing.

imt-pm and Oracle Process Management and Notification

imt-pm is integrated with Oracle Process Management and Notification during the installation of Web Conferencing (Real-Time Collaboration).

Service Availability Monitoring

With Oracle Real-Time Collaboration, you can check service availability on an instance. Sometimes, it might not be enough for the processes to be up. It is also important that the instance be able to provide the conference services.

Using imtctl, determine service availability by invoking:

```
imtctl> runtests -testlist mtgtest,voiceconvtest,docconvtest
```

This command checks if the instance on which the command is executed is providing the following services:

Conference service—A conference test mimics the behavior of an end-user going through the same flow of Oracle Web Conferencing Application as the user, and starting an instant conference. It makes sure that the conference starts successfully on one of the available collaboration servers in an instance. It then joins another client in the same conference, and makes sure that the new client also gets the conference state consistent with the host of the conference. It finally ends the test conference. This test thus makes sure that conference service is available in a particular Real-Time Collaboration instance. If this test fails, make sure that one or more mxs, and one or more collaboration servers are up in this instance (using `imtctl> getState`).

Voice conversion service—This test verifies that voice streaming service is available by mimicking the behavior of a collaboration server for connecting to an available voice conversion server, and making sure that T1 line is up, there are available voice channels, and the server is able to stream audio. If this test fails, then either no voice conversion server has been setup to service this instance, or the ones configured are not available.

Document conversion service—This test verifies document conversion service availability by mimicking document conversion flow. It uploads a test document, and then tries converting that document using the document conversion servers available to the instance. If this test fails, then either no document conversion server has been setup to service this instance, or the ones configured are not available.

Real-Time Collaboration also publishes interfaces for service availability monitoring that can be integrated into any monitoring infrastructure. See "Real-Time Collaboration Interfaces" on page 7-5 for more information.

The service availability tests can be invoked periodically by a monitoring infrastructure, and the results of the tests can be plugged into an alert management system.

For more details about `runtests` see Chapter , "Testing and Monitoring the System".

Component Monitoring

With Oracle Real-Time Collaboration, you can get monitoring data for the key components in the system. The monitoring data includes the following.

For each imt-collab on an instance, you can obtain the following data:

- number of active conferences
- number of active users
- total memory
- used memory
- total conferences since the time the process was started

Use `getMonitorStats` on the instance to get these statistics, or use the `-i` option to direct the command towards a different instance.

For the Voice Conversion Server on that instance, you can obtain the following data:

- total number of channels
- currently used channels
- bad channels
- T1 line status

Use `getMonitorStats` on the instance to get these statistics, or use the `-i` option to direct the command toward a different instance.

Oracle Real-Time Collaboration publishes interfaces for getting this data in XML. This can then be invoked periodically by a monitoring infrastructure for historical analysis. Use getMonitorStats with the publish option.

```
imtctl> getMonitorStats -publish true
```

For details about `getMonitorStats`, see "Testing and Monitoring the System" on page 10-13.

Conference Monitoring

The Oracle Web Conferencing Application provides Web pages to monitor the conferences that are currently running on the system. The Monitor tab in the Web Conferencing Application provides access to this functionality.

> **Note:** Only users with the businessmonitor or businessadmin role can use the Monitor tab. See "Setting User Roles" on page 10-10 for details about setting user roles.

The Current Conference Status page under the Monitor tab lists all the conferences that are currently running on the system. For each conference, it provides the conference ID, conference title, host name, conference type, site, start time, total attendees, and current status of the conferences. It also provides a Conference Details link, so you can see detailed information about each conference that is currently running.

In the Conference Details pages, you can find information about the attendees participating in that conference, including details about each attendee. Conference level details like current collaboration mode, voice start time, and so on can also be found in these pages. You can also see the conference logs for this conference from this page.

Instances Status Page

The Instances Status Page is available from the **System** tab in the Web Conferencing Application Web pages.

> **Note:** Only users with the businessadmin role can use the Monitor tab. See "Setting User Roles" on page 10-10 for details about setting user roles.

The Instances report lets you further monitor the activity of instances and components running within the Web Conferencing system. You can click to expand or contract hierarchical lists of instances and their components. The report also shows the results of tests on instances, the progress of active conferences, the properties set for the system, all sites created for the system, and details about currently-running Web Application sessions.

Configuration Tests

The following tests confirm that some important post-installation steps, like e-mail configuration, which are required for the proper functioning of the Real-Time Collaboration system have been successfully completed, and serve as a verification tool for the administrator. These tests need to be run only once after an installation, and need not be run periodically. See "Testing and Monitoring the System" on page 10-13 for details about running the tests.

E-mail Configuration Test

This test checks whether the post-installation step required for Oracle Web Conferencing e-mail invitations to work has been performed by the administrator. For e-mail invitations to work, enterprise SMTP host and port should be specified through Oracle Web Conferencing properties. The test makes sure that these properties are set and the SMTP server is accessible from the Oracle Real-Time Collaboration mid-tier.

Proxy Configuration Test

This test checks whether the post-installation step required for Oracle Web Conferencing cobrowsing to work has been performed by the administrator. This test needs to succeed only if the mid-tier needs an enterprise server to access internet. This test makes sure that the properties required for proxy configuration are set, and that the proxy server is accessible from the Oracle Real-Time Collaboration mid-tier.

MX Configuration Test

To support internet users where clients need to connect from behind their corporate proxies, mod_imeeting needs to be enabled and some properties need to be set in the Real-Time Collaboration system. This test makes sure that the required properties are set, and that mod_imeeting has been enabled to handle HTTP/HTTPS requests to connect to Mx.

Repository (Database) Access Test

Because all Real-Time Collaboration services depend on the database as the single-point of information and backbone, a test is provided to verify access to the Real-Time Collaboration Repository. This test makes sure that database is available (that is, it is accessible and responding to queries).

Configuration Status

The Configuration status report is available from the **System** tab in the Web Conferencing Application Web pages.

> **Note:** Only users with the businessadmin role can use the Monitor tab. See "Setting User Roles" on page 10-10 for details about setting user roles.

The Configuration report shows the current settings for all services for this Web Conferencing system. It displays the host name, type of service, deployment information (whether the host is in the intranet, Internet, or the DMZ), instance location if any, operating system, and hardware information: platform, CPU, and total memory. The report also has an **Edit** button to let you interactively edit some of the properties for each server listed, and a **Delete** button to let you delete the service.

Real-Time Collaboration Interfaces

This section discusses Real-Time Collaboration monitoring interfaces that can be plugged into any monitoring infrastructure. There are two types of interfaces:

- *Servlet interface*—for monitoring service availability
- *imtctl interfaces*—for getting monitoring data about key Real-Time Collaboration components.

Servlet Interfaces

Oracle Web Conferencing includes a suite of functionality tests that are designed to verify the availability of major Web Conferencing functionalities. The tests cover Real-Time Collaboration Repository connectivity, Voice Conversion Server availability, Document Conversion Server availability, application availability, and the ability to launch an instant conference.

The **ImtTestServlet** lets these tests be run by any monitoring infrastructure. ImtTestServlet acts as an adapter that makes all Real-Time Collaboration tests available as a Web application to HTTP-based Web application monitoring. The servlet is accessible to standard HTTP-based monitoring clients. Its results are designed for automated analysis. ImtTestServlet is the most flexible choice for low-effort integration of existing monitoring systems with the Real-Time Collaboration functionality tests.

Inputs to the Servlet

As an HTTP servlet, ImtTestServlet gets all of its input information from HTTP requests sent by the client. The servlet accepts parameters either through the URL query string or via a POST body. Either way, the same parameters are supported. Input parameters control which tests are run and the information that is returned in the case of success or failure.

Table 7–1 Inputs to the ImtTestServlet

Name of Test	Options	Default	Comments
alltests	true, false	false	Run all tests supported by the servlet (other test selection parameters are ignored).
mtgtest	true, false	false	Run the end-to-end conference test to verify that conference functionality is available.
voiceconvtest	true, false	false	Run the Voice Conversion Server test to verify that voice support is available for conferences.
docconvtest	true, false	false	Run the Document Conversion Server test to verify that document conversion is available.
dbtest	true, false	false	Run the Real-Time Collaboration Repository test to verify that it is available.
errorcode	any valid HTTP response code	500	Sets the HTTP response code sent when any of the selected tests fail.
successcode	any valid HTTP response code	200	Sets the HTTP response code sent when all of the selected tests succeed.
errormsg	Any string	null	Message included in the response body when any of the selected tests fail. (Note: the response body may include additional text, as well.)
successmsg	Any string	"Test(s) successful."	Message included in the response body when all of the selected tests succeed. (Note: the response body may include additional text, as well.)

Output from the Servlet

The ImtTestServlet provides its results via an HTTP response. Results either report that all of the selected tests succeeded or that some test failed. If multiple tests are selected via input parameters, the result provides no details about which particular tests failed. Furthermore, no messages associated with the failure are returned.

The result of the test or tests is reflected in both the HTTP response code and, optionally, in a static string returned in the response body. It is anticipated that these two mechanisms will be sufficient to allow integration products to detect whether the test succeeded or failed.

Because the test servlet reports only aggregate results when running multiple tests, Oracle Corporation advises that the servlet be called multiple times independently when fine-grained failure detection is desired (one request for each test to be run). However, you might choose to have a single indicator of system health, in which case, running all of the tests in a single request is an optimal configuration.

Samples

The following examples are designed to illustrate some of the input parameter combinations that might be useful. The examples are templated to account for variable host names, ports, and application root contexts.

http://<host>:<port>/<root-context>/servlet/ImtTestServlet

This confirms that the test servlet has been properly installed. It does not run any tests.

http://<host>:<port>/<root-context>/servlet/ImtTestServlet?alltests=true

This runs all of the tests returning the standard error (500) and success (200) codes.

http://<host>:<port>/<root-context>/servlet/ImtTestServlet?mtgtest=true&errorcode=404

This runs only the end-to-end conference test and returns 404 if the test fails.

http://<host>:<port>/<root-context>/servlet/ImtTestServlet?mtgtest=true&voiceconvtest=true&errormsg=mtgorvoicefailed

This runs the conference and voice tests and reports a custom message on failure, in addition to a standard 500 response code.

Limitations

The ImtTestServlet is currently limited in the following way. Because the servlet runs in an OC4J_imeeting, the inaccessibility of the OC4J_imeeting will prevent granular detection of failures in other Real-Time Collaboration components, such as the Web Conferencing Servers and Document Conversion Servers. However, this limitation is minor, because OC4J_imeeting is the gatekeeper for all Web Conferencing services. Inaccessibility of OC4J_imeeting is equivalent to inaccessibility of all Web Conferencing services from a client's perspective.

8

Reports

Oracle Real-Time Collaboration supports various reporting capabilities, including e-mailed reports and usage trend information available within the Oracle Web Conferencing Application. Some aspects of these features require post-installation configuration, such as including sender's and receiver's e-mail addresses. See Chapter 4, "Post-Installation" for details about configuration.

Usage Report

The Usage report is available from the **Reports** tab in the Web Conferencing Application Web pages.

> **Note:** Only users with the businessmonitor or businessadmin role can use the Reports tab. See "Setting User Roles" on page 10-10 for details about setting user roles.

The Usage report is designed to give high-level Web Conferencing usage metrics to business administrators and business managers. The report contains data which is based on one of the following two metrics:

- Time (minutes/hours)
- Usage numbers (count of conferences, users, and so on)

The Key Performance Indicators (KPI) are then summarized with a trending pattern that displays the count and percentage change over the previous time period. The report gives business administrators and business managers an idea of what collaborative modes and features are used, top users, and line of business information. The usage report is computed for a weekly period. You can also view historical data by entering the relevant dates.

The following metrics are captured in the usage report.

Conference minutes summary:

- Total user minutes
- Total conference minutes
- Total voice minutes
- Total recorded minutes
- Total playback minutes
- Host rating

- Average conference length
- Longest conference (minutes)

Conference usage summary:

- Conferences
- Voice conferences
- Total users
- Total recordings
- Total playbacks
- Users in conferences
- Voice users
- Average users per conference
- Largest conference (users)

User registration and login details:

- Unique users for the week and to date
- New registrations for the week and total registrations to date

Top 10 lists of Users and Sites

In addition to the weekly KPIs table, the conference minutes and usage details are displayed in a time series of week, month, quarter, and year. The weekly average over the highest period is also computed.

Feedback Reports

The Feedback report is available from the **Reports** tab in the Web Conferencing Application Web pages.

> **Note:** Only users with the businessmonitor or businessadmin role can use the Reports tab. See "Setting User Roles" on page 10-10 for details about setting user roles.

The feedback report is designed to give high-level system performance feedback via user ratings to the business administrator and to business managers. The report contains data based on performance ratings given by conference hosts at the end of every conference.

The host of the conference can give a comment (raw text feedback) in addition to rating a conference excellent, good, or poor. Optionally, the host can even decide to not rate the conference, in which case the conference is classified with no response. The Key Performance Indicators are then summarized with a trending pattern displaying the count and percentage change over the previous time period.

Below the KPIs table, the top ten conferences with poor, good, excellent, and no response ratings with user comments are listed. These are grouped into four individual tables. Only conferences with host comments are listed in the tables. This convenient format allows administrators to proactively respond to the users who rate the conferences poor.

Uptime Reports

Oracle Real-Time Collaboration provides tests that can be used by monitoring infrastructures to periodically monitor Web Conferencing for service availability. This data can then be used produce uptime reports. See "Real-Time Collaboration Interfaces" on page 7-5 for more information about the tests and how to integrate them with your monitoring infrastructure.

9

Web Conferencing Sites

Oracle Web Conferencing lets you easily configure a single deployment instance to meet the requirements of various lines of business, by creating individual sites and customizing system, application, and conference level properties for those sites.

Web Conferencing sites are created by the business administrator. (See "Setting User Roles" on page 10-10 for details about assing a user the business administrator role.) For every site created, the integrating application or line of business uses a unique authentication token along with the site ID to communicate with the Web Conferencing Application. Creating a site provides the following benefits:

- Customized system, application, and conference properties.
- Support for custom integrated flows which can bypass the Web Conferencing Application.
- Support for custom Web Conferencing Console behavior based on the properties set for the site.
- Site-level reports which give the same depth and breakdown of data as the global Web Conferencing site reports.
- Custom user interfaces, including a separate Oracle Web Conferencing prelogin application page with full support for listing public conferences only for the specific site.
- Custom branding to uniquely identify the integrating site/line of business.

The global site is pre-created and cannot be deleted. By default, all users belong to site "iMeeting," which is a global site. However, a user can belong to more than one site.

Integrating Applications with a Site

The Real-Time Collaboration system uses the concept of site to provide a customized “view” of the Oracle Web Conferencing deployment, for each of the integrating applications. In an enterprise, a single deployment of the Real-Time Collaboration system can be used by various integrated applications supporting lines of business, such as a sales department, a support department, etc. To support the integrating applications, you create sites with unique site IDs.

An application in an enterprise can integrate with Oracle Web Conferencing in two ways:

- It can programmatically invoke Web Conferencing functions using Oracle Real-Time Collaboration Integration Service calls or,
- If the application has a browser-based interface, it can invoke the Oracle Web Conferencing Application pages directly.

In either case, a site ID enables the Real-Time Collaboration system to provide a customized service for each integrating application.

Creating a Site

The Oracle Web Conferencing administrator creates a site by providing a unique site name, display name, and description for the site. To create a site, the administrator does the following:

1. Log in with an administrator account (the account must have been assigned the businessadmin role as described in "Setting User Roles" on page 10-10).
2. Click the **Sites** tab.
3. Click the **Create Site** button.
4. Create a site, assigning it a Site Name (required) and optional display description.
5. Click **Apply**

The Real-Time Collaboration system then registers the site and provides the administrator with a site ID and an authentication token, both of which are system-generated. An application that wants to integrate with the Real-Time Collaboration system can use the new site ID to make the appropriate calls.

Using a Site ID

For Oracle Real-Time Collaboration Integration Service calls, the site ID is passed as one of HTTP header fields in the HTTP request to invoke the service. For any application Web page that provides a link to the Oracle Web Conferencing Application page, it can provide the site ID as a URL parameter. For example, the following URL can be used to provide a customized view of Oracle Web Conferencing for site 123456.

```
http://<hostname>:<port-number>/imtapp/app/prelogin.uix?siteID=123456
```

Entering this URL into a browser returns a page that has the look and feel and contents specified for site 12345 using any Web Conferencing properties, as described in "Customizing Site Properties" on page 9-2. Any conferences created from the application pages will have the attributes defined for conferences for that site.

Customizing Site Properties

After you have created a site, you can then set up site-specific properties to control the interaction of the integrating site with Oracle Web Conferencing. For all properties that you do not change, the values of the corresponding default Web Conferencing site apply. See Chapter 5, "Configuration" for descriptions of properties that can be set at the site level.

There are two broad classes of properties that affect how users interact with the site:

- Application Properties: Change the look and feel of the Oracle Web Conferencing Application pages. Examples of such properties are the co-branding name that appears at the top of each application page, the e-mail address of the Contact Us page, the number of rows displayed for each table in the application page, and so on.
- Conference Properties: Change the attributes of any conference, whether it is invoked from the application pages or from the integration service calls. Examples of such properties include the Startup (collaboration) Mode of the conference, the availability of chat during the conference, and so on.

Figure 9–1 Site Property Inheritance

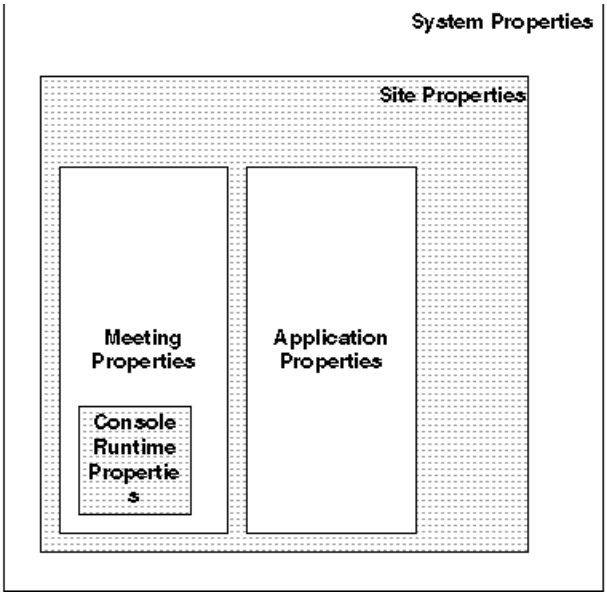


Figure 9–1, "Site Property Inheritance" shows how the various property values in Oracle Web Conferencing are inherited and set. When a site is created, it inherits the default values of all the properties from the system settings. For each site the administrator can set various conference-level and application-level properties to override the default values. Some of the conference-level properties can be changed from inside the Web Conferencing Console when the conference is running. For example, the default setting for the system is to set the start mode of each conference to Cobrowse mode. When a site is created, this default value is inherited for that site. The site administrator can override this default value by changing the `StartupMeetingMode` property to, for example, Desktop Sharing mode.

10

imtctl Command Line Utility

The **imtctl** utility provides a command-line interface for administering and configuring the Real-Time Collaboration system. The utility supports a variety of commands to let you start and stop Real-Time Collaboration processes, view their current state, and configure your Web Conferencing system.

Running imtctl

The imtctl utility is available on all platforms under `$ORACLE_HOME/imeeting/bin`. There are two ways to run imtctl commands interactively. You can also create scripts to run imtctl commands.

- Enter the entire `imtctl` command at the command line. This executes one command at a time. For example:

  ```
  > $ORACLE_HOME/imeeting/bin/imtctl setProperty -system true -pname
  "StartupMeetingMode" -pvalue "DesktopSharing"
  ```

- Enter `imtctl` without any options. This opens the utility in interactive command mode. See "Using imtctl in Command-line Mode" on page 10-1 for details.

The following sections describe how to use imtctl in interactive mode, and how to run imtctl scripts.

Using imtctl in Command-line Mode

To start imtctl in command-line mode, enter `imtctl` in an operating system shell or command window, without any options:

```
$ORACLE_HOME/imeeting/bin> imtctl
imtctl>
```

You can then enter as many imtctl commands as you wish:

```
imtctl> setProperty -siteId 1024536 -pname "MeetingStartUrl" -pvalue
"www.oracle.com"
imtctl> setProperty -siteId 1024536 -pname "MeetingEndUrl" -pvalue
"www.oracle.com"
imtctl>
```

To display a list of supported commands, enter `help`:

```
imtctl> help
```

To get specific instructions for a particular command, enter the command followed by the `-help` option:

```
imtctl> getState -help
```

You stop command-line mode by entering `exit`:

```
imtctl> exit
```

imtctl Scripts

You can write simple scripts for the imtctl utility, consisting of imtctl commands and optional comments.

Oracle Web Conferencing is shipped with a set of standard scripts you can use to perform typical post-installation configuration tasks. These scripts reside in the `$ORACLE_HOME/imeeting/bin/scripts` directory and end with the extension `.imt`. Each script has sample lines to indicate the values that must be set before running the script. Run the scripts using standard input redirection as in the following example:

```
$ORACLE_HOME/imeeting/bin> imtctl < scripts/sample.imt
```

You can create new scripts as needed. The scripts can contain any supported imtctl commands, plus comment lines (any line that begins with #) and the special `echo [on/off]` command for echoing commands executed by scripts.

Setting System, Instance, Component, and Site Values

As discussed in Chapter 5, you can use imtctl to configure an entire system or particular instances or components. "List of imtctl Commands" on page 10-3 describes the syntax for all of the imtctl commands, including the options that are available. The following table shows the possible values for any the scope-related options; you may use these values with any command that includes one of these options.

Table 10–1 Valid Values for System, Instance, or Component Options

imtctl Option	Scope it Affects	Valid Value(s)
`-system`	Entire system	true
`-i`	An instance	The full instance name: *instance_name*.machinename.domainname, where *instance_name* is a name you assign as you install the instance. Use `imtctl listInstances` to display the IDs for all instances for this system, or click the **System** tab to display all instances. Example: `myinst.machine.us.oracle.com`
`-cid`	A single component, by its ID	A number assigned by the Real-Time Collaboration system to this component when you install it. Use `imtctl listComponents` to display the IDs for all components in the current instance, or click the **System** tab to display all components and their IDs for your system.

Table 10–1 (Cont.) Valid Values for System, Instance, or Component Options

imtctl Option	Scope it Affects	Valid Value(s)
`-cname`	Any component of a specific name	Any of the following names: `imt-collab`: Web Conferencing server `imt-docconv`: Document Conversion server `imt-mx`: Multiplexer (mx) `imt-pm`: Web Conferencing Process Monitor `imt-voiceconv`: Voice Conversion server
`-ct`	Any component of a specific type	Any of the following types: `clbsvr`: Web Conferencing server `docconv`: Document Conversion server `imtpm`: Web ConferencingProcess Monitor `mxcomm`: Multiplexer (mx) `voiceconv`: Voice Conversion server
`-siteID`	A single site, by its ID	A number assigned by the Real-Time Collaboration system to this site when you create it. Click the **Sites** tab to display all sites and their IDs for your system.

List of imtctl Commands

Following is a brief summary of all imtctl commands, listed alphabetically.

Table 10–2 imtctl Commands

Command	Use to	See
`addSysDialin`	Add a dial-in number for voice conferencing	"Setting Conference Dial-In Information"
`deleteSysDialin`	Delete a voice conferencing dial-in number	"Setting Conference Dial-In Information"
`exit` or `quit`	Exit imtctl command-line mode	"Getting Help and Quitting imtctl"
`getMonitorStats`	Get monitoring statistics	"Testing and Monitoring the System"
`getPids`	Get identifiers for current Web Conferencing processes	"Testing and Monitoring the System"
`getProperties`	Get the current value of properties at a specified scope	"Setting and Displaying Properties"
`getProperty`	Get the current value of any property	"Setting and Displaying Properties"
`getState`	Determine the current status of Web Conferencing components	"Testing and Monitoring the System"
`getSysDialins`	Display all dial-in numbers currently set	"Setting Conference Dial-In Information"
`help`	Display a list of available commands	"Setting Conference Dial-In Information"
`listComponents`	Display current properties for Web Conferencing components	"Listing System Information"

Table 10–2 (Cont.) imtctl Commands

Command	Use to	See
`listInstances`	Display current properties for Web Conferencing instances	"Listing System Information"
`modifyRole`	Set or change roles for any Web Conferencing user	"Setting User Roles"
`runTests`	Run status tests on the Web Conferencing system	"Testing and Monitoring the System"
`setProperty`	Set properties to configure the Web Conferencing system	"Setting and Displaying Properties"
`start`	Start a Web Conferencing instance	"Starting and Stopping an Instance"
`stop`	Stop a Web Conferencing instance	"Starting and Stopping an Instance"
`versions`	List version information for an instance	"Listing System Information"

The following sections discuss the imtctl commands, grouped as follows:

- Starting and Stopping an Instance
- Listing System Information
- Setting and Displaying Properties
- Setting User Roles
- Setting Conference Dial-In Information
- Testing and Monitoring the System
- "Getting Help and Quitting imtctl"

Starting and Stopping an Instance

You use the `start` and `stop` commands to start any instance or component processes.

> **Note:** Although the `start` and `stop` commands let you start and stop individual components for administrative purposes, this document strongly recommends that in general you start or stop only a complete instance whenever possible.

start

Start the current instance, or individual components in the current instance. See Table 10–1 for the valid values for instance and component options.

- start

  Start all the components in the current instance. This is the recommended way to start an instance, because it handles all startup dependencies.

- start -i *instance-name*

  Start all components in a specific instance.

- start -cid *component-ID*

Start the component with the given ID.

- start -cname *component-name*

  Start the component(s) with the given name.

- start -ct *component-type*

  Start the component(s) with the given type.

Example 1: Starting a Component with a Specific ID

To start a component with ID 10001 in the current instance, enter:

```
imtctl> start -cid 10001
```

> **Note:** The component with the ID you enter must be running on the instance where you enter the imtctl command.

Example 2: Starting a Component with a Specific Name

To start a component named imt-collab in the current instance, enter:

```
imtctl> start -cname imt-collab
```

> **Note:** The component with the name you enter must be running on the instance where you enter the imtctl command.

stop

Stop the current instance, or individual components in the current instance. See Table 10–1 for the valid values for instance and component options.

- stop

  Stop all the components in the current instance. This is the recommended way to stop an instance, because it stops all components in the instance in appropriate order.

- stop -i *instance-name*

  Stop all components in a specific instance.

- stop -cid *component-ID*

  Stop the component with the given ID.

- stop -cname *component-name*

  Stop the component(s) with the given name.

- stop -ct *component-type*

  Stop the component(s) with the given type.

Example 1: Stopping a Component with a Specific ID

To stop a component with ID 10001, enter:

```
imtctl> stop-cid 10001
```

> **Note:** The component with the ID you enter must be running on the instance where you enter the imtctl command.

Example 2: Stopping a Component with a Specific Name

To stop any component named imt-collab, enter:

```
imtctl> stop -cname imt-collab
```

> **Note:** The component with the name you enter must be running on the instance where you enter the imtctl command.

Listing System Information

Three informational commands, `listInstances`, `listComponents`, and `versions`, let you display details about parts of your system.

listInstances

List all the instances in the Real-Time Collaboration system.

```
imtctl> listInstances
INSTANCE NAME              HOST NAME             IMT HOME
instance1.oracle.com       host1.oracle.com      C:/core/imeeting
instance2.oracle.com       host2.oracle.com      /u02/90200b/imeeting
```

listComponents

List the components in any Real-Time Collaboration instance. See Table 10–1 for the valid values for component and instance options.

- listComponents

  List information about all components on the current instance.

- listComponents -i *instance-name*

  List all the components for a particular instance.

- listComponents -cid *component_ID*

  List the component with the given ID.

- listComponents -cname *component-name* [-i *instance-name*]

  List the component with the given name. If -i is used, the component with the given name in the given instance is listed.

- listComponents -ct *component-type* [-i *instance-name*]

  List the component with the given type. If `-i` is used, the component with the given type in the given instance is listed.

Example 1: Listing Components in the Current Instance

To list the components in the current instance, enter:

```
imtctl> listComponents
ID      NAME            TYPE       DESCRIPTION               NUM_PROCS
10006   imt-collab      clbsvr     Collaboration Server      4
10008   imt-pm          imtpm      iMeeting Process Monitor  1
10009   imtctl          imtctl     iMeeting CmdLine Control  1
10007   OC4J_imeeting   oc4j       OC4J                      1
10005   imt-mx          mxcomm     Multiplexer               1
```

Example 2: Listing Components for a Specific Instance

To list the components in a different instance (instance1.company.com), enter:

```
imtctl> listComponents -i instance1.company.com
ID      NAME             TYPE       DESCRIPTION                NUM_PROCS
10003   imt-voice        voiceconv  Voice Conversion Server    1
10004   imt-converter    docconv    Document Conversion Server 1
```

Example 3: Listing Components with a Specific ID

To list the component with the ID 10007, enter:

```
imtctl> listComponents -cid 10006
ID      NAME             TYPE       DESCRIPTION                NUM_PROCS
10007   OC4J_imeeting    oc4j       OC4J                       1
```

Example 4: Listing Components with a Specific Type

To list the components in the current instance with the type `clbsvr`, enter:

```
imtctl> listComponents -ct clbsvr
ID      NAME             TYPE       DESCRIPTION                NUM_PROCS
10006   imt-collab       clbsvr     Collaboration Server       4
```

versions

List the software versions for an instance or components. See Table 10–1 for the valid values for instance and component options.

- versions [-i *instance-name*]

  List versions for all components in the current instance or a specified instance.

- versions -cid *component-ID*

  List the version for a specific component, by its ID.

- versions -cname *component-name* [-i *instance-name*]

  List the versions for all components with a specific name, and optionally for a specific instance.

- versions -ct *component-type* [-i *instance-name*]

  List the versions for all components with a specific type, and optionally for a specific instance.

Example 1: List Versions for the Current Instance

To list the software versions running on all components in the current instance, enter:

```
imtctl> versions
ID        NAME             TYPE            VERSION
10006     imt-collab       clbsvr          2.0.4.3.0 Rel20430_135
10008     imt-pm           imtpm           2.0.4.3.0 Rel20430_135
10007     OC4J_imeeting    oc4j            2.0.4.3.0 Rel20430_135
```

Example 2: List Versions for a Component on a Specific Instance

To list the software version of the document conversion server on myinstance.oracle.com, enter:

```
imtctl> versions -ct docconv -i myinstance.oracle.com
ID        NAME             TYPE            VERSION
10010     imt-docconv      docconv         2.0.4.3.0 Rel20430_135
```

Setting and Displaying Properties

You configure the Web Conferencing system by setting properties. The `setProperty`, `getProperty`, and `getProperties` commands let you set and display property details.

See Table 5–2, " Web Conferencing Properties" in Chapter 5, "Configuration" for a list of all of the properties you can set. That chapter also provides specific syntax for each property.

setProperty

Set a property to configure a Real-Time Collaboration system, instance (default) component, and/or site. See Table 10–1 for the valid values for system, instance, component, and site options.

- imtctl> setProperty -pname *property-name* -pvalue *property-name*

  Set the value for a property at instance level.

- imtctl> setProperty -cid *component-id* -pname *prop_name* -pvalue *prop_value*

  Set the value of a property for the component with the given ID.

- imtctl> setProperty -cname *component-name* [-i *instance-name*] -pname *prop_name* -pvalue *prop_value*

  Set the value of a property for the components of the given name in the current instance. If -i is used, it sets the property for the components of the given name in the given instance.

- imtctl> setProperty -ct *component-type* [-i *instance-name*] -pname *prop_name* -pvalue *prop_value*

  Set the value of a property for the components with the given type in the current instance. If -i is used, it sets the property for the components with the given type in the given instance.

- imtctl> setProperty -system *true* -pname *property_name* -pvalue *property_value*

  Set the value for a property at the system level.

- imtctl> setProperty -siteId *site-id* -pname *property_name* -pvalue *property_value*

  Set the value for a property at site level.

- imtctl> setProperty -pname *property-name* -pvaluenull true|false

  Sets the value of a property to null if -pvaluenull is true. Use this to unset a property so that it has a null value. Some properties interpret null in a special manner. For example, the GlobalWebhost property, once configured, cannot be unset unless set to a value of "null." By default, the value for -pvaluenull is *false*, in which case a -pvalue is required.

Example 1: Setting a Property for an Instance

To set the ApacheWebHost property for the current instance to imeeting4.company.com, enter:

```
imtctl> setProperty -pname ApacheWebHost -pvalue imeeting4.company.com
```

Example 2: Setting a Property for a System

To set the system-wide log-level to SEVERE, enter:

```
imtctl> setProperty -system true -pname LogLevel -pvalue SEVERE
```

See Chapter 5, "Configuration" for more examples.

getProperty

Get a property at a specified scope. Oracle Real-Time Collaboration supports system, instance (default), component, and site scopes. See Table 10–1 for the valid values for system, instance, component, and site options.

- imtctl> getProperty -pname *property_name*

  Get the value for the property for this instance.

- imtctl> getProperty -cid *component-id* -pname *property_name*

  Get the value of the property for the component with the given ID.

- imtctl> getProperty -cname *component-name* [-i *instance-name*] -pname *property_name*

  Get the value of the property for the components of the given name in the current instance. If -i is used, it gets the property for the components of the given name in the given instance.

- imtctl> getProperty -ct *component-type* [-i *instance-name*] -pname *property_name*

  Get the value of the property for the components with the given type in the current instance. If -i is used, it gets the property for the components with the given type in the given instance.

- imtctl> getProperty -system true -pname *property_name*

  Get the value for the property at the system level.

- imtctl> getProperty -siteId *site-id* -pname *property_name*

  Get the value for all the properties at site level for the specified site

Example 1: Getting a Property for the Current Instance

To get the ApacheWebHost property for the current instance, enter:

```
imtctl> getProperty -pname ApacheWebHost
The effective value for instance myinstance.ap079sun.us.oracle.com of the property
"ApacheWebHost" is "imeeting4.company.com"
```

Example 2: Getting a Property for the System

To display whether the system has SSL security enabled (it is possible to create secure meetings), enter:

```
imtctl> getProperty -system true -pname GlobalMeetingSSLSupportEnabled
The effective value at system scope of the property
"GlobalMeetingSSLSupportEnabled" is "true"
```

For a list of all possible properties, see Chapter 5, "Configuration".

getProperties

Displays an alphabetized list of all properties at the specified scope and higher. You can display properties for the system and for an instance , component, or site. See Table 10–1 for the valid values for system, instance, component, and site options.

- imtctl> getProperties

  Display the values for the properties for this instance and for the system.

- imtctl> getProperties -ct *component-type* [-i *instance-name*]

Display the value of all the properties for the component with the given type in the current instance. If -i is used, it gets the property for the components with the given type in the given instance. It will return properties for the component, instance and system.

- imtctl> getProperties -system true

  Display only system properties.

- imtctl> getProperties -siteId *site-id*

  Display the value for all the properties for the specified site and the system.

Example 1: Displaying the Properties for the System Only

To display only system-level properties, enter:

```
imtctl> getProperties -system true
AllSitesID="100"
ApplicationContainerName="RTC"
ApplicationEntryName="RTCApplication"
CheckConnectionTimePeriod="15"
ConsolePrivacyEnabled="true"
CorpImageName="oracle_ocs_crop.gif"
DateFormat="dd-MMM-yyyy h:mm a"
...
```

> **Note:** The command displays all properties set for your system, including those set by default or through the Web Conferencing graphical user interface. The properties you might normally wish to set using `setProperty` to configure your system are discussed in Chapter 5, "Configuration".

Example 2: Displaying the Properties for a Component by Type

To display the properties of all collaboration server (`clbsvr`) components on this instance, as well as instance and system properties, enter:

```
imtctl> getProperties -ct clbsvr
AdditionalLocationsServed="(null)"
AdminEmail="imeeting@us"
AllSitesID="100"
ApacheProtocolSecure="true"
ApacheTunnelHost="machine1.us.oracle.com"
ApacheTunnelPort="443"
ApacheWebHost="machine1.us.oracle.com"
ApacheWebPort="80"
ApacheWebSecurePort="443"
...
```

Setting User Roles

As described in "User Management" on page 2-10, the Web Conferencing system uses the Oracle Internet Directory store to identify users. You assign *roles* to these users to allow them access to various features of the Web Conferencing system. You use the `modifyRole` command to assign the roles.

modifyRole

Sets a user's role, to control what features of the Web Conferencing system the user may access.

modifyRole -username *username* -rolename *rolename*

Assigns a role to a named user. The user's name is that shown in the Oracle Internet Directory, such as `john.smith@oracle.com`. The role can be any of the following:

- **enduser**: Can use any of the standard Web Conferencing features, such as scheduling a meeting, uploading mneeting materials, and viewing meeting archives.
- **businessmonitor**: Can use standard Web Conferencing features, and can also view the Monitor and Reports tabs to monitor current conferences and see reports regarding conference history, usage, and user's feedback about conferences.
- **businessadmin**: Can use any of the previous-listed features, and can also view the Site and System tabs to create and manage Web Conferencing sites and view statistics about all instances and components of the system.

> **Note:** Only a user with businessadmin privileges can set other users' roles. As discussed in Chapter 4, "Post-Installation", you must set at least one businessadmin user for a Web Conferencing installation.

Example: Setting a Business Administrator Role

To set a jane.roe@oracle.com to be a business administrator, enter:

```
imtctl> modifyRole -username jane.roe@oracle.com -rolename businessadmin
```

Setting Conference Dial-In Information

If you use streaming voice conferences, the addSysDialin, deleteSysDialin, and getSysDialins commands let you manage the dial-in numbers for any of your phone conference vendors. See Chapter 5, "Configuration" for more details and examples of how you might set dial-in information.

addSysDialin

Sets a named dial-in with a pre-programmed number that users can choose from a drop-down list when scheduling a conference. You can enter a complete number with conference ID and password for a recurring phone conference, or enter a template number with text indicating where users need to substitute their own information. When users select this dial-in, they can edit the number in the text field.

- addSysDialin -name *dialin-name* -sequence *dialin-sequence* [-default true]

  Adds a dial-in with a specified name and number (sequence). If -default is set to true, this dial-in is the default for all users of the Web Conferencing system.

Note: The Voice Conversion server handles the actual dialing of the number. If your company has a prefix that must be entered to get an outside line, you set that prefix using the VoiceDialInPrefix property. Do **not** enter the prefix in the dial-in. See "Setting Up Document and Voice Conversion Servers for Web Conferencing" on page 4-9 for more details.

Example 1: Setting a Default Dial-in for a System

To set a dial-in named Standard Phone Conference that can be used by all users as a default, enter:

```
imtctl> addSysDialin -name "Standard Conference" -sequence "18005551234,,<Conf
ID>#<Passcode>#,,,,,,,,,,#" -default true
New system dialin created with id = 21994
```

The Web Conferencing system automatically assigns the dial-in an ID number.

In the previous example, users can remove the text in angle brackets and type in their specific conference ID and passcode for a particular conference. The commas (,) cause dialing to pause for one second. The # symbol represents the user pressing the # button to complete entering an ID or password (typically required by most phone conference vendors). If your phone conference system includes additional message to which the user must respond by pressing #, you can include commas to pause during the message and a # for response.

Example 2: Setting a Dial-in for a Recurring Conference

To set a dial-in for a recurring conference with ID 80904 and passcode 56221, enter:

```
addSysDialin -name "Sales Force Mtg" -sequence
"18005551234,,80904#56221#,,,,,,,,,,#"
```

getSysDialins

Displays the currently-set dial-ins.

```
getSysDialins
ID     NAME                  SEQUENCE
21994  Standard Conference   18005551234,,<Conf ID>#<Passcode>#,,,,,,,,,,#
21998  Sales Force Mtg       18005551234,,80904#56221#,,,,,,,,,,#
```

deleteSysDialin

Deletes an existing dial-in number. You can delete the dial-in by entering either its name, or its ID. For example, to delete a conference named Standard Conference, enter:

```
imtctl> deleteSysDialin -name "Standard Conference"
Standard Conference has been deleted
```

To delete a conference with the ID 21994, enter:

```
imtctl> deleteSysDialin -id 21994
Dialin with id 21994 has been deleted
```

Testing and Monitoring the System

The getState, getPids, `getMonitorStats`, and runTests commands let you view the state of various components and processes, and run quick tests on instances or components.

getState

Shows the state of components in any instance. See Table 10–1 for the valid values for component and instance options.

- imtctl> getState [-i *instance-name*]

  Show the state of all the components in the current instance. Use -i to show the state of components in an instance with the given name.

- imtctl> getState -cid *component-ID*

  Show the state of the component with the given ID.

- imtctl> getState -cname *component-name* [-i *instance-name*]

  Show the state of the components of the given name in the current instance. If -i is used, it shows state of the components with the given name in the given instance.

- imtctl> getState -ct *component-type* [-i *instance-name*]

  Show state of the components with the given type in the current instance. If -i is used, it shows state of the components with the given type in the given instance.

Example: Getting the State of All Components

To get the state of all the components in the current instance, enter:

```
imtctl> getState
ID       NAME           TYPE         STATUS     NUM_PROCS
10001    imt-collab     clbsvr       UP         4
10000    mx             mxcomm       UP         1
```

getPids

Display the process identifiers for all running processes. See Table 10–1 for the valid values for instance and component options.

- getPids [-i *instance-name*]

  Display the process identifiers for either the current instance, or a specified instance if -i is used.

- getPids -cid *component-id*

  Display the process identifiers for a specific instance, by its ID number.

- getPids -cname *component-name* [-i *instance-name*]

  Display the process identifiers for components by name. If -i is used, displays the identifiers for the named components in the named instance.

- getPids -ct *component-type* [-i *instance-name*]

  Display the process identifiers for components by type. If -i is used, displays the identifiers for the specified components in the named instance.

Example 1: Display All Process Identifiers for an Instance

To display all the process identifiers for the current instance, enter:

```
imtctl> getPids
ID       NAME           COMPONENT TYPE    PIDS
10006    imt-collab     clbsvr            12650
10006    imt-collab     clbsvr            912
10006    imt-collab     clbsvr            12549
10006    imt-collab     clbsvr            31966
10008    imt-pm         imtpm             25921
10007    OC4J_imeeting  oc4j              15585
10005    imt-mx         mxcomm            26003
```

Example 2: Displaying Process Identifiers for Components on an Instance

To display the process identifiers for any document conversion servers on instance `myinstance.oracle.com`, enter:

```
imtctl> getPids -ct docconv -i myinstance.oracle.com
ID       NAME          COMPONENT TYPE     PIDS
10010    imt-docconv   docconv            3020
```

getMonitorStats

Dispays monitoring statistics for a system, instance, or component. See Table 10–1 for the valid values for system, instance and component options. See Chapter 7, "Monitoring" for more information about monitoring components.

- getMonitorStats [-i *instance-name*]

  Display statistics for the current instance or, if -i is used, for a named instance.

- getMonitorStats -cid *component-id*

  Display statistics for a particular component in this instance, by the component ID number.

- getMonitorStats -cname *component-name* [-i *instance-name*]

  Display statistics for all components of a particular name in this instance or, if -i is used, in a named instance.

- getMonitorStats -ctype *component-type* [-i *instance-name*]

  Display statistics for all components of a particular type in this instance or, if -i is used, in a named instance.

- getMonitorStats -system true

  Display statistics for all components in the system.

- getMonitorStats -publish true

  Display statistics for all components in this instance and display them in XML format, for use in integrating with other applications. The `-publish` option can be used with options that specify system, site, component, or instance.

The displayed results vary depending on the types of components. The results list may include any of the following.

Statistics for collaboration server components:

- `TMTGS`: Number of conferences since the process was started
- `CMTGS`: Number of active conferences
- `CLTS`: Number of active users
- `TMEM`: Total memory allocated for this component

- UMEM: Total memory currently used by this component

Statistics for voice conversion server components:

- AVAIL: Whether the voice conversion server is available
- T1LINE: Whether a T1 line is available
- IN-USE: Number of voice channels currently in use
- IDLE: Number of voice channels idle
- BAD: Number of bad voice channels

Example 1: Displaying Statistics for an Instance

To display statistics for all components in a specific instance, enter:

```
intctl> getMonitorStats -i instance1.oracle.com
Instance - instance1.oracle.com:
Component Name: imt-collab, Component Type: clbsvr
SERVICE_NAME                              TMTGS CMTGS  CLTS  TMEM     UMEM
clbsvr:instance1.oracle.com.imt-collab.01     1     2     4,708K   4,285K
clbsvr:instance1.oracle.com.imt-collab.11     0     0     4,624K   3,836K
clbsvr:instance1.oracle.com.imt-collab.20     0     0     8,644K   3,457K
```

These statistics show there is currently one conference being run on imt-collab.01, with two attendees.

Example 2: Displaying Statistics for a System

To display statistics for all components and instances in this system, enter:

```
intctl> getMonitorStats -system true
Instance - instance1.oracle.com:
Component Name: imt-collab, Component Type: clbsvr
SERVICE_NAME                              TMTGS CMTGS  CLTS  TMEM     UMEM
clbsvr:instance1.oracle.com.imt-collab.01     1     2     4,708K   4,285K
clbsvr:instance1.oracle.com.imt-collab.11     0     0     4,624K   3,836K
clbsvr:instance1.oracle.com.imt-collab.20     0     0     8,644K   3,457K
Instance - instance2.oracle.com:
Component Name: imt-voiceconv, Component Type: voiceconv
SERVICE_NAME                                    AVAIL T1LINE IN-USE  IDLE   BAD
voiceconv:instance2.oracle.com.imt-voiceconv.0 true  true   0       12     0
```

These statistics show there is currently one meeting being run on imt-collab.01 with two attendees, and there are 12 channels on the voice conversion server available on a T1 line but none currently in use.

runTests

Runs Real-Time Collaboration tests on a specific instance or all instances in the system. See Table 10–1 for the valid values for instance options. See Chapter 7, "Monitoring" for more information about using the tests.

- runTests [-i *instance-name*]

  Run tests in the current instance. Use -i to run all the tests in an instance with the given name.

- runTests -testlist *test-name*, *test-name*, *test-name...*

  The possible tests are:

  - apptest database connectivity test

- dbtest database connectivity test
- docconvtest document conversion service test
- emailtest e-mail configuration test
- mtgtest conference service test
- mxmodtest multiplexer/mod_imeeting configuration test
- proxytest proxy configuration test
- voiceconvtest voice conversion service test

- runTests -system true

  Run tests on all the instances in the system.

- runTests -publish true

  Run tests on the current instance with the output displayed in structured XML tags. This option can be used in scripts to provide output to services integrated with Web Conferencing.

- runTests -v true

  Run tests on the current instance with the verbose option to display any messages regarding test failures.

- runTests -cluster true

  Run the conference service tests on a cluster of instances. If this instance is not part of a cluster, the test is only run on the current instance.

Example 1: Testing an Instance

After installing an instance, enter the following to see if the instance is configured properly and working. The `runTests` command will run all appropriate tests for components installed on this instance.

```
imtctl> runTests
Instance - myinstance.oracle.com
TESTNAME              SUCCESS
mtgtest               true
voiceconvtest         true
docconvtest           true
dbtest                true
apptest               true
mxmodtest             true
proxytest             true
emailtest             true
```

Example 2: Running a Single Test

To run the conference test alone, invoke the following

```
imtctl> runTests -testlist mtgtest
Instance - myinstance.oracle.com
TEST NAME             SUCCESS
mtgtest               true
```

Example 3: Running Multiple Tests With the Verbose Option

To run multiple tests with verbose display, enter:

```
imtctl> runTests -testlist mtgtest,emailtest,modtest -v true
Instance - myinstance.oracle.com
```

```
TEST NAME            SUCCESS   MESSAGE
mtgtest              true
emailtest            true
modtest              false     Unknown test
```

In the example, one option was entered incorrectly (`modtest` instead of `mxmodtest`), so the message displayed by the `-v` option reports the problem.

Getting Help and Quitting imtctl

The `help` command displays help about all commands available in imtctl. You can exit the imtctl command line interface by entering `exit` or `quit`.

help

Displays the list of imtctl commands.

```
imtctl> help
Commands are:
start - Start a specified component or complete instance.
stop - Stop a specified component or complete instance.
getstate - Gets the state of a specified component or complete instance.
...
```

exit or quit

Leaves the imtctl command-line interface. You can use either `exit` or `quit`.

```
> $ORACLE_HOME/imeeting/bin/imtctl
imtctl>
imtctl> exit
>
```

11

Troubleshooting

After Oracle Web Conferencing has been successfully configured and is working, there are two kinds of problems that require an administrator's intervention:

- *User complaints*—Some users are unable to use Oracle Web Conferencing, or they are unable to log in, etc.
- *Alerts*—An administrator receives alerts after integrating Oracle Real-Time Collaboration monitoring into the company's monitoring infrastructure.

In addition, an administrator could experience problems while configuring the system.

For suggested steps to resolve these problems and for other common guidelines, see the Troubleshooting documents and other reference documentation posted at http://www.oracle.com/technology/products/ortc/index.html

A

Time Zones

The following table contains a list of time zones supported by Oracle Real-Time Collaboration. See "Property to Configure Time Zones" on page 5-34 for details about setting the default time zone for a Web Conferencing system.

Table A–1 Real-Time Collaboration Time Zones

Internal Name	External User Visible Name
Pacific/Pago_Pago	(-11:00) Pago Pago
Pacific/Honolulu	(-10:00) Hawaii
America/Anchorage	(-09:00) Alaska
America/Vancouver	(-08:00) Canada Pacific Time
America/Los_Angeles	(-08:00) US Pacific Time
America/Tijuana	(-08:00) Tijuana
America/Edmonton	(-07:00) Canada Mountain Time
America/Denver	(-07:00) US Mountain Time
America/Phoenix	(-07:00) Arizona
America/Mazatlan	(-07:00) Mazatlan
America/Winnipeg	(-06:00) Canada Central Time
America/Regina	(-06:00) Saskatchewan
America/Chicago	(-06:00) US Central Time
America/Mexico_City	(-06:00) Mexico City
America/Guatemala	(-06:00) Guatemala
America/El_Salvador	(-06:00) El Salvador
America/Managua	(-06:00) Managua
America/Costa_Rica	(-06:00) Costa Rica
America/Montreal	(-05:00) Canada Eastern Time
America/New_York	(-05:00) US Eastern Time
America/Indianapolis	(-05:00) East Indiana
America/Panama	(-05:00) Panama
America/Bogota	(-05:00) Bogota
America/Lima	(-05:00) Lima

Table A–1 (Cont.) Real-Time Collaboration Time Zones

Internal Name	External User Visible Name
America/Halifax	(-04:00) Canada Atlantic Time
America/Puerto_Rico	(-04:00) Puerto Rico
America/Caracas	(-04:00) Caracas
America/Santiago	(-04:00) Santiago
America/St_Johns	(-03:30) Newfoundland
America/Sao_Paulo	(-03:00) Sao Paulo
Atlantic/Azores	(-01:00) Azores
Etc./UTC	(00:00) Universal Time
UTC	(00:00) Universal Time
Atlantic/Reykjavik	(00:00) Reykjavik
Europe/Dublin	(00:00) Dublin
Europe/London	(00:00) London
Europe/Lisbon	(00:00) Lisbon
Africa/Casablanca	(00:00) Casablanca
Africa/Nouakchott	(00:00) Nouakchott
Europe/Oslo	(+01:00) Oslo
Europe/Stockholm	(+01:00) Stockholm
Europe/Copenhagen	(+01:00) Copenhagen
Europe/Berlin	(+01:00) Berlin
Europe/Amsterdam	(+01:00) Amsterdam
Europe/Brussels	(+01:00) Brussels
Europe/Luxembourg	(+01:00) Luxembourg
Europe/Paris	(+01:00) Paris
Europe/Zurich	(+01:00) Zurich
Europe/Madrid	(+01:00) Madrid
Europe/Rome	(+01:00) Rome
Africa/Algiers	(+01:00) Algiers
Africa/Tunis	(+01:00) Tunis
Europe/Warsaw	(+01:00) Warsaw
Europe/Prague	(+01:00) Prague Bratislava
Europe/Vienna	(+01:00) Vienna
Europe/Budapest	(+01:00) Budapest
Europe/Sofia	(+02:00) Sofia
Europe/Istanbul	(+02:00) Istanbul
Europe/Athens	(+02:00) Athens
Asia/Nicosia	(+02:00) Nicosia
Asia/Beirut	(+02:00) Beirut

Table A–1 (Cont.) Real-Time Collaboration Time Zones

Internal Name	External User Visible Name
Asia/Damascus	(+02:00) Damascus
Asia/Jerusalem	(+02:00) Jerusalem
Asia/Amman	(+02:00) Amman
Africa/Tripoli	(+02:00) Tripoli
Africa/Cairo	(+02:00) Cairo
Africa/Johannesburg	(+02:00) Johannesburg
Europe/Moscow	(+03:00) Moscow
Asia/Baghdad	(+03:00) Baghdad
Asia/Kuwait	(+03:00) Kuwait
Asia/Riyadh	(+03:00) Riyadh
Asia/Bahrain	(+03:00) Bahrain
Asia/Qatar	(+03:00) Qatar
Asia/Aden	(+03:00) Aden
Africa/Khartoum	(+03:00) Khartoum
Africa/Djibouti	(+03:00) Djibouti
Africa/Mogadishu	(+03:00) Mogadishu
Asia/Dubai	(+04:00) Dubai
Asia/Muscat	(+04:00) Muscat
Asia/Yekaterinburg	(+05:00) Yekaterinburg
Asia/Tashkent	(+05:00) Tashkent
Asia/Calcutta	(+05:30) India
Asia/Novosibirsk	(+06:00) Novosibirsk
Asia/Almaty	(+06:00) Almaty
Asia/Dacca	(+06:00) Dacca
Asia/Krasnoyarsk	(+07:00) Krasnoyarsk
Asia/Bangkok	(+07:00) Bangkok
Asia/Saigon	(+07:00) Vietnam
Asia/Jakarta	(+07:00) Jakarta
Asia/Irkutsk	(+08:00) Irkutsk
Asia/Shanghai	(+08:00) Beijing, Shanghai
Asia/Hong_Kong	(+08:00) Hong Kong
Asia/Taipei	(+08:00) Taipei
Asia/Kuala_Lumpur	(+08:00) Kuala Lumpur
Asia/Singapore	(+08:00) Singapore
Australia/Perth	(+08:00) Perth
Asia/Yakutsk	(+09:00) Yakutsk
Asia/Seoul	(+09:00) Seoul

Table A–1 (Cont.) Real-Time Collaboration Time Zones

Internal Name	External User Visible Name
Asia/Tokyo	(+09:00) Tokyo
Australia/Darwin	(+09:30) Darwin
Australia/Adelaide	(+09:30) Adelaide
Asia/Vladivostok	(+10:00) Vladivostok
Australia/Brisbane	(+10:00) Brisbane
Australia/Sydney	(+10:00) Sydney Canberra
Australia/Hobart	(+10:00) Hobart
Asia/Magadan	(+11:00) Magadan
Asia/Kamchatka	(+12:00) Kamchatka
Pacific/Auckland	(+12:00) Auckland

Glossary

DMZ

The "demilitarized zone" is a subnetwork located between a company's corporate intranet and the Internet.

Document Conversion Server

Document Conversion Server enables the Web Conferencing Application to automatically convert documents for the Document Presentation mode.

Oracle Web Conferencing Application

The set of components used to administer the Web Conferencing system. It consists of the application Web pages, Web listener, and the Document Conversion Server.

Oracle Web Conferencing Console

The console where real-time conferences take place.

Load Balancer

A very fast network device which can distribute Web requests to a large number of physical servers. The purpose of a load balancer (LBR) is to provide a single published address to the client browsers, and, in the case of Real-Time Collaboration, provide multiple Real-Time Collaboration Core Components mid-tiers which actually service the requests, based on the distribution of the requests done by the LBR.

Web Conferencing Server

The backend server that manages all of the runtime state associated with a conference. Clients never directly communicate with the Web Conferencing Server. Instead, clients communicate with the HTTP Listener, which uses mod_imeeting to communicate with the appropriate Web Conferencing Server.

Voice Conversion Server

The Voice Conversion Server enables listen-only voice streaming by capturing voice stream from the Dialogic card, encoding it using GSM codec, and streaming it to the Web Conferencing Server for recording or listen-only voice support.

Index

O

P

Q

R

S

T

U

V

W