Oracle Video Server™

Content Administrator's Guide

Release 3.0

February, 1998

Part No. A59688-01

ORACLE®

Enabling the Information Age™

Oracle Video Server Content Administrator's Guide

Part No. A59688-01

Release 3.0

Printed in the U.S.A

Primary Author: John Dowden

Contributors: Kim Bartlett, Chris Bedford, Mike Christian, Lesley Kew, Ming Lee, Mine Ogura, Mason Ng, Dave Pawson, Matt Prather, Pat Ritto, Michael Short, Karl Stewart, Pari Tajiki, Manish Upendran, Dan Weaver, John Zussman

Contents

Send Us Your Comments ix

Preface xi

1 Introduction

The Oracle Video Server Content Model 1-2
Content Layers 1-2
Running OVS without a Database 1-3
Content Management Interfaces 1-4

2 Creating Source Video

Identifying the Projects 2-2
Sample Applications 2-2
Broadband Digital Video Applications 2-2
Enterprise Digital Video Applications 2-3
Low bit rate Digital Video Applications 2-3
Planning and Purchasing Recording Equipment 2-4
Digital Video Equipment 2-4
Analog Video Equipment 2-4
Recording Your Video Project 2-5
Editing Your Video Project 2-6
Video Editing Software 2-6

3 Encoding Content

Video Encoding Methods 3-2
Employ a Video Production Studio 3-2
Purchase a Capture / Encoding Station 3-3
Hardware 3-3
Software 3-3
Audio / Video Gear 3-4
Codecs, Containers, and Frameworks 3-4
Codecs (Compressors / Decompressors) 3-5
Containers (File Formats) 3-6
MPEG-1 System and MPEG-2 Transport 3-6
Oracle Streaming Format (OSF) 3-6
Raw Key Frame (RKF) 3-6
How Different Video Clients Support Different Encoding Formats 3-7
Encoding MPEG Files 3-8
What is MPEG? 3-8
MPEG Standards 3-8
How the Oracle Video Server Supports MPEG-1 3-10
How the Oracle Video Server Supports MPEG-2 3-10
Before You Encode 3-11
Encode a Test File First 3-12
MPEG Encoding Samples 3-13
MPEG-1 Encoding Tips 3-14
Elementary Streams 3-14
I-Frame Regularity 3-14
Non-sequential PTS and SCR 3-14
Sequence Header Manipulation 3-14
MPEG-2 Encoding Tips 3-15
Elementary Streams 3-15
MPEG-1 or MPEG-2 Video Layer 3-15
Non-sequential PCR 3-15
NPT Descriptors 3-15
PAT and PMT Manipulation 3-15
PCR Placement 3-16
Sequence Header and I-Frame Regularity 3-16

Sequence Header Manipulation 3-16
Program Specification Information 3-16
Encoding AVI Files 3-17
What is AVI? 3-17
How the Oracle Video Server Supports AVI 3-18
Encoding AVI Files for Network Computer (NC) Clients 3-19
Before You Encode 3-19
AVI Encoding Samples 3-21
AVI Encoding Tips 3-22
Encoding Rate 3-22
Key frames 3-22
Encoding WAV Files 3-23
Examples 3-23

4 Loading and Managing Digital Content

Content Administration Tasks 4-2
Loading Content Files 4-4
Loading Content with Video Server Manager 4-5
Loading Content Using **mdstar** 4-5
Loading Content Using **mdscopy** 4-6
Loading Content Using FTP 4-7
Copying Content Files 4-9
Copying Content with Video Server Manager 4-9
Copying Content Using **mdscopy** 4-9
Copying Content Using Passive FTP 4-10
Archiving Content Files 4-11
Archiving MDS Content Files Using **mdstar** 4-12
Archiving MDS Content Files Using FTP 4-12
Deleting Content Files 4-15
Deleting MDS Content Files Using **mdsdelete** 4-15
Verifying the Integrity of an MPEG-2 Transport File 4-16
Registering Content Files 4-17
Registering Content with Video Server Manager 4-18
Registering Content Using **vstag** 4-18
Registering Content Using **vsmkosf** 4-20

Registering Content Using **vsgentag** 4-21
Managing Registered Content 4-23
Displaying Tag File Content 4-23
Modifying Tag Files 4-26
Updating the Content Registry in the Oracle Database 4-27
Deleting Obsolete Content Information from the Oracle Database 4-28
Registering Tagged Content with the Oracle Database 4-30
Managing Logical Content with Video Server Manager 4-31
Creating Logical Content 4-32
Modifying Logical Content 4-32
Removing Logical Content 4-33
Managing Clips with Video Server Manager 4-33
Creating Clips 4-34
Modifying Clips 4-34
Removing Clips 4-34
Managing Physical Content with Video Server Manager 4-35
Viewing Physical Content Descriptions 4-35
Removing Physical Content Descriptions 4-35
Managing Programs with Video Server Manager 4-36
Creating a Program Schedule with VSM 4-36
Modifying a Program Schedule with VSM 4-37
Removing a Program Schedule with VSM 4-38

A Oracle Media Data Store Naming Conventions and FTP Usage Notes

Naming MDS Volumes and Files A-2
Referencing Files on the MDS Filesystem A-2
Referencing Files on a Host Filesystem A-3
FTP Usage Notes A-4
Referencing Volumes with MDS FTP A-4
Supported File Types A-4
Sending the allo Command A-4

Index

Examples

4–1 Loading Content from Tape to the MDS Using **mdstar** 4-6
4–2 Loading a File from a Host File System to an MDS Volume Using **mdscopy** 4-6
4–3 Loading Multiple Files from a Host File System to an MDS Volume Using **mdscopy**... 4-7
4–4 Loading Content From a UNIX Host to the MDS Using FTP 4-7
4–5 Copying a File from one MDS Volume to Another 4-9
4–6 Copying a File Between Two Video Servers Using Passive FTP 4-10
4–7 Archiving Content from the MDS to Tape Using **mdstar** 4-12
4–8 Archiving All Files in an MDS Volume into a Single File 4-12
4–9 Archiving MDS Content Files Using FTP 4-13
4–10 Deleting a Content File from the MDS Using **mdsdelete** 4-15
4–11 Verifying the Integrity of a Single MPEG-2 Transport File 4-16
4–12 Verifying the Integrity of Multiple MPEG-2 Transport Files 4-16
4–13 Registering an MPEG-1 File with **vstag** 4-18
4–14 Registering an MPEG-2 File with **vstag** 4-18
4–15 Registering an OSF File with **vstag** 4-19
4–16 Registering a Corrupted Content File with **vstag** 4-19
4–17 Registering Multiple MPEG Files with **vstag** 4-19
4–18 Loading and registering a Content File in One Command 4-19
4–19 Converting an AVI File to an OSF File 4-20
4–20 Converting a WAV File to an OSF File 4-20
4–21 Converting Multiple AVI Files to OSF Files 4-20
4–22 Registering an MPEG-1 File with **vsgentag** 4-22
4–23 Registering an Motion-JPEG File with **vsgentag** 4-22
4–24 Displaying the Tag File Contents of an MPEG-1 File 4-23
4–25 Displaying the Tag File Contents of an OSF File 4-25
4–26 Updating File Names with **vstagpatch** 4-26
4–27 Updating the Rate Control Prohibitions with **vstagpatch** 4-26
4–28 Deleting a Logical Content Title Using **vscontdel** 4-29
4–29 Deleting a Logical Content Title and Tag File Using **vscontdel** 4-29
4–30 Deleting a Logical Content Title, Tag File, and Content File Using **vscontdel** 4-30
4–31 Registering a File with the Database 4-30
4–32 Registering Multiple Files with the Database 4-30
4–33 Rewriting a File's Database Entry 4-30
A–1 Referencing a File in the MDS Filesystem A-2
A–2 Referencing a File in the Working Directory A-3
A–3 Referencing a File Relative to the Working Directory A-3
A–4 Referencing a File Relative to the Parent Directory A-3

Send Us Your Comments

Oracle Video Server Content Administrator's Guide, Release 3.0

Part No. A59688-01

Oracle Corporation 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 chapter, section, and page number (if available). You can send comments to us in the following ways:

- electronic mail - omsdoc@us.oracle.com
- FAX - 650.506.7615 Attn: Oracle Media Server Documentation Manager
- postal service:
  Oracle Corporation
  Oracle Media Server Documentation
  500 Oracle Parkway, Mailstop 6OP5
  Redwood Shores, CA 94065
  U.S.A.

If you would like a reply, please give your name, address, and telephone number below.

Preface

As the content administrator for an Oracle Video Server system, you need to be familar with all aspects of content creation, management, and deployment. This guide addresses the primary tasks that you might be responsible for as the content administrator for your company. These tasks include:

- identifying potential uses for digital video
- recording and editing content
- encoding content
- loading and managing your digital content

This preface provides the following information:

- Intended Audience
- Structure
- Related Documents
- Conventions

Intended Audience

This manual is intended for anyone who is interested in creating content for and managing content from the Oracle Video Server system.

This guide does *not* describe how to develop server and client applications or third-party management tools that could be used with the Oracle Video Server. To learn how to develop these types of applications, refer to the *Oracle Video Server Developer's Guide* and the *Oracle Video Client Developer's Guide*.

To learn more about system planning and configuration issues or to learn more about a specific command-line utility, refer to the *Oracle Video Server Administrator's Guide and Command Reference*.

This guide assumes that you are familiar with the Microsoft Windows and UNIX operating systems.

Structure

This manual contains the following chapters and appendix:

Chapter 1 **Introduction:** describes the different layers of content available in the Oracle Video Server environment.

Chapter 2 **Creating Source Video:** describes how to identify beneficial uses for video in your company and how to film and edit your video in preparation for digital conversion.

Chapter 3 **Encoding Content:** describes how to encode your digital content for use with the Oracle Video Server system.

Chapter 4 **Loading and Managing Digital Content:** describes how to load and manage your content with the Oracle Video Server system.

Appendix A **Oracle Media Data Store Naming Conventions and FTP Usage Notes:** describes how the Oracle MDS treats volume and file names and provides FTP usage notes.

Related Documents

Refer to the *Oracle Video Server Road Map* to find documents related to this release of the Oracle Video Server system.

Conventions

This section describes the command and platform conventions used in this guide.

The following conventions are 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.
	Buttons, icons, and dialog box names appear in boldface type.
	Menu commands appear in boldface type with the name of the menu and the name of the command separated by a vertical bar, **menu | command**.
< >	Angle brackets enclose user-supplied names.
[ ]	Brackets enclose optional clauses from which you can choose one or none.
$	The dollar sign represents the DIGITAL CommandLanguage prompt in OpenVMS and the Bourne shell prompt in Digital UNIX

Your Comments Are Welcome

We value and appreciate your comments as an Oracle user and reader of the manuals. As we write, revise, and evaluate our documentation, your opinions are the most important input we receive. Our Send Us Your Comments form is at the front of the manual, at the end of the table of contents. We encourage you to use this form to tell us what you like and dislike about this manual or other Oracle manuals. If the form is not available, please use the following address or FAX number.

Oracle Video Server Documentation Manager
Oracle Corporation
500 Oracle Parkway
Box 6OP5
Redwood Shores, CA 94065
U.S.A.
FAX: 650-506-7615
Email: omsdoc@us.oracle.com

1

Introduction

You can use the Oracle Video Server (OVS) system to incorporate digital video into a variety of applications, such as interactive training-on-demand, point-of-sale kiosks, Web sites, and multimedia catalogs. To make best use of the digital content that you store in your OVS system, however, you must understand how the OVS stores, organizes, and manages content.

This chapter provides the following topics:

- The Oracle Video Server Content Model
- Content Management Interfaces

The Oracle Video Server Content Model

The Oracle Video Server (OVS) offers both physical and logical control of all content that is stored in the OVS system. To harness the flexibility and control that this content model offers, you must understand the different layers of content that are available in the OVS environment.

Content Layers

The OVS content model consists of the following layers:

- **Content Files:** files that the OVS stores and delivers to clients. Content files can contain video and/or audio and are created through a process called encoding.

  For more information on the types of content files that the OVS supports, refer to "Codecs, Containers, and Frameworks" in Chapter 3.

- **Tag Files:** a physical file that stores metadata and header information about a given piece of content, such as the file's name, format, bit rate, and size. Tag files also store information about individual video frames, which is not stored in the database. Tag files must have an **.mpi** extension.

- **Physical Content:** a database object that contains the same header and metadata information found in a tag file.

- **Clips:** a logical excerpt from a content file. Each clip corresponds to only one content file. Clips map to a specific start and stop position (in seconds) within a content file.

- **Logical content:** a collection of video clips that are played in a pre-defined sequence.

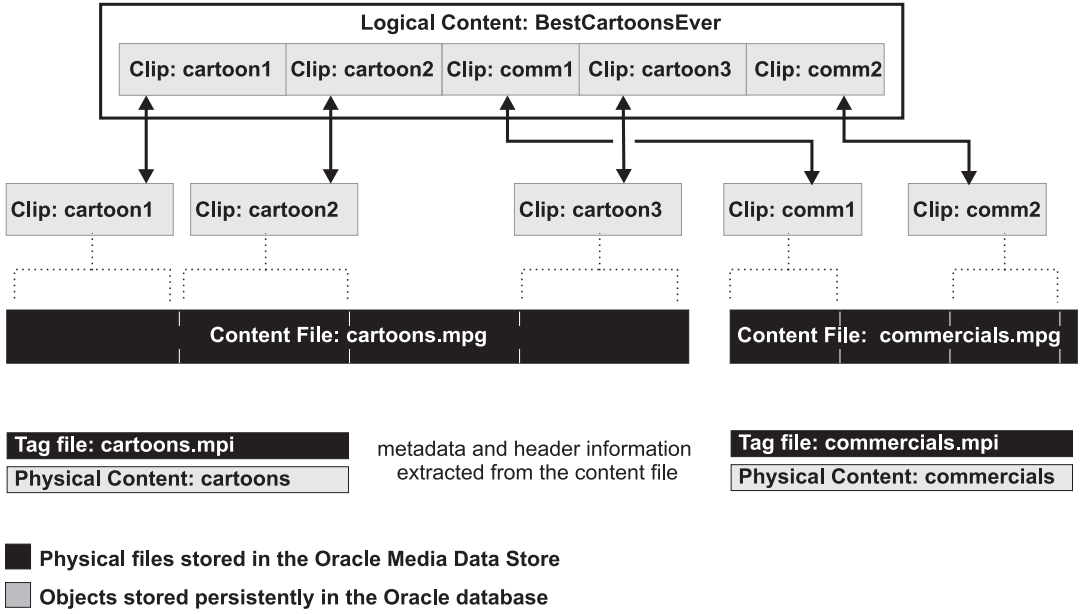
Content files and tag files are physical data files which are stored in the Oracle Media Data Store (MDS). For each content file that you load into the MDS, you must create a tag file and store it in the MDS with the content file.

In contrast, logical content, clips, and physical content are all objects that are stored persistently in the Oracle database. Using Oracle Video Server Manager, you can create, edit, and remove these objects, thereby protecting the physical data that is stored in the MDS.

In the OVS environment, the process of creating tag files and creating the logical content, clip, and physical content objects in the database is called **registering**. For more information on how to register content, refer to "Registering Content Files" in Chapter 4.

Figure 1–1 illustrates the relationship between the different OVS content layers.

Figure 1–1 Oracle Video Server Content Layers



Running OVS without a Database

If you are running OVS without a database, the content model changes. When OVS is invoked without a database, the **content service** obtains content data directly from the tag file headers stored in the MDS. Thus, there are no clips or logical content titles in a system that does not have a database.

Content Management Interfaces

The Oracle Video Server system provides two content management interfaces:

- Oracle Video Server Manager console, a graphical, Java-based application
- the OVS command-line utilities

Each interface is designed for different types of video server administrators and, therefore, offer different levels of content management support. When used in tandem, these interfaces provide complete control over all the physical and logical content stored in the OVS system.

When Should I Use Oracle Video Server Manager?

Oracle Video Server Manager is designed to make content management easier. VSM streamlines complex command-line utilities, offering simple, fill-in-the-blank dialog boxes. For example, with Video Server Manager, you can load an AVI file into the Oracle MDS, convert the AVI file to an OSF file, and register the OSF file with the Oracle MDS and database in a single step.

When Should I Use the Command-line Interface?

Some content administrators require more control over how the Oracle Video Server executes utilities. For this reason, you can still use the Oracle Video Server command-line utilities for your specific needs. The command-line interface is intended for expert users and should not be used without a full understanding of how the utilities work.

Refer to Chapter 4, “Loading and Managing Digital Content” for a complete list of content administration tasks that you can perform with the Oracle Video Server system.

For a more detailed description of any of the command-line utilities used in this document, refer to the *Oracle Video Server Administration and Reference Guide*.

2

Creating Source Video

This chapter describes how to identify beneficial uses for video in your company and how to film and edit your video in preparation for digital encoding.

This chapter provides the following information:

- Identifying the Projects
- Planning and Purchasing Recording Equipment
- Recording Your Video Project
- Editing Your Video Project

Identifying the Projects

Before filming new video, buying stock footage, or editing existing video, determine how and where you want to make video available by asking yourself the following questions:

- How can I use video to gain a competitive advantage?
- How can I use video to help my employees be more productive?
- How can I use video to help my customers?
- How can I use video to improve communication?

Answers to these types of questions will help you identify the projects for which digital video deployment is critical.

In the process of identifying these projects, you might discover that your company has an existing video library that can be digitized and put online. To learn how to encode these videos, refer to Chapter 3, "Encoding Content".

Sample Applications

You can use digital video to create a variety of applications, such as interactive training-on-demand, product announcements, point-of-sale kiosks, pay-per-view services, and Web sites.

This section provides examples of how you might use digital video in three different types of networking environments:

- Broadband
- Enterprise
- Low Bit Rate

Broadband Digital Video Applications

The broadband environment is characterized by high-speed delivery rates of extremely large content files. Typically, this environment requires the use of a dedicated **set-top box**. Video applications in a broadband environment include:

- A pay-per-view service that delivers video from a central location to thousands of subscribers.
- Subscription-based service that delivers a digitally-recorded event, such as a blacked-out athletic game.

Enterprise Digital Video Applications

In an enterprise environment, files are not usually as long, or as large, as those used in broadband environments, but there is still a need for many concurrent viewers over a corporate LAN, WAN, or intranet. Video applications in an enterprise environment include:

- Delivery of corporate training videos at a time convenient to the employee, rather than as a single scheduled event with many employees at a time.
- Corporate announcements and product announcements to the sales force
- Information stored in a "corporate repository" and available for retrieval as needed—for example, detailed step-by-step instructions for enrolling in your company's 401(k) program, instructions for installing a piece of equipment, or an introductory campus/facility tour.

Low bit rate Digital Video Applications

The low bit rate environment is typically geared toward delivery of digital video over the Internet. Most people still connect to the Internet via modems with a relatively slow rate of delivery (low bit rate). Video applications that are particularly effective in this environment include:

- Advertising. Many commercial locations now have Web sites that feature information digital video to acquaint you with their product.
- News. You can provide a video-based, news-delivery Web site to which people can subscribe.

Planning and Purchasing Recording Equipment

If you want to create digital video, you need to plan the purchase of the appropriate equipment. (Your company might already have much of the equipment listed here.) When creating new video material, you have two choices:

- **Digital video**: Digital video cameras enable you to capture video directly to the computer. In this scenario, the video is never stored in an analog form.
- **Analog video**: The traditional, analog video camera enables you to record video on tape, which can then converted to a digital form using special video capture equipment.

Of course, higher quality equipment yields higher quality videos.

Digital Video Equipment

Digital video cameras enable you to capture video directly to the computer. Some of the digital video cameras currently available include:

- Sony DSR-200
- Hitachi MP-EG1A (MPEG camera)
- JVC GR-DVX Digital CyberCam
- Sharp VL-DX10U Digital Viewcam

Analog Video Equipment

When choosing your analog video equipment, consider the following:

- **Use the best quality camera and tape available to you. Your choices, in order of decreasing quality, include:**
  - Digital tape: D1, D2, or D3
  - Betacam or Betacam SP (professional quality)
  - Hi-8
  - S-VHS
  - VHS or 8mm
- **Video cameras provide features that can help you produce better source video. We recommend using a camera with the following features:**
  - 400 scan-line image resolution (S-VHS or Hi-8) or better (Betacam or digital tape). VHS and standard 8mm support only 250 lines.

  - image stabilization
  - manual control over exposure and focus
  - hi-fi audio (stereo) for S-VHS; AFM or PCM for Hi-8
  - time code
  - headphone and microphone jacks
- **Accessories can make filming easier and help you produce better video. We recommend having some or all of the following:**
  - headphones
  - carrying case
  - tripod
  - extra batteries and tape
  - lens and head cleaners
  - lighting equipment
  - wireless microphone

Recording Your Video Project

To optimize your video for encoding and compression, follow a few simple rules when you record your video projects:

- Move in for close-ups; zoom, tilt, or pan slowly, if at all. Rapid movements do not digitize well.
- Film your subject as closely as possible, with occasional medium-distance shots for variety. Avoid long-distance shots, especially with rapidly-moving subjects.
- When possible, frame your subjects against solid-colored, unchanging backgrounds. This approach digitizes and compresses well.
- Keep your film clips as short as possible, within reason. Record continuous video sequences that can be digitized into a single, continuous movie file.
- Avoid recording at slow tape speeds. Slower tape speeds produce grainier source video.
- When using Hi-8, use 60-minute tape. The 120-minute tapes are 25% thinner and may break.
- Use a tripod. Unsteady camera movements do not digitize well.

- Use the dominant light source in a mixed lighting environment to manually calibrate your camera's white balance control.
- Use a microphone positioned close to your speaker or audio source. Avoid using the video camera microphone unless there are no other sound sources in the room.
- Always wear headphones if you are capturing audio with your video. This allows you to monitor the audio input as it's being recorded.

Editing Your Video Project

If you plan to edit your video, you must do so before you encode it. Performing digital editing after encoding will damage valuable encoding data that the video server and client need to play your video successfully.

When editing your video, consider the following:

- Digital content comsumes a lot of disk space. To save space and editing time, capture only the clips you plan to use.
- End a clip as soon as the action finishes.
- Do not create scenes that are less than three seconds long. Shorter scenes do not provide enough time for the viewer's eyes to adjust.
- Avoid using jump cuts. Jump cuts use two separate clips to present the same scene. This editing technique is jarring and looks unprofessional. To avoid video jumps, insert a different scene between the two similar scenes. For example, you could insert a scene of a person talking about the scene. Alternatively, you can use the transition features in your editing software to create a more fluid transition. Since the audio data does not jump, you can use the audio from both clips.
- Make the credits fade on or pop on rather than crawl. Crawling credits do not compress as well during encoding. If credits must crawl, they should do so slowly and use large type.

Video Editing Software

For basic video editing, several video editing software packages are available:

- Adobe After Effects (**http://www.adobe.com**)
- Adobe Premiere (**http://www.adobe.com**)
- Asymetrix Digital Video Producer (**http://www.asymetrix.com**)

- Corel Lumiere Suite (**http://www.corel.com**)
- MetaTools Final Effects (**http://www.metatools.com**)

After you have edited your video to your satisfaction, you must encode it in a format which is appropriate for your delivery medium, as described in Chapter 3, “Encoding Content”.

3

Encoding Content

This chapter describes how to encode your source media content for use with the Oracle Video Server system and includes the following information:

- Video Encoding Methods
- Codecs, Containers, and Frameworks
- How Different Video Clients Support Different Encoding Formats
- Encoding MPEG Files
- Encoding AVI Files
- Encoding WAV Files

Video Encoding Methods

Video encoding captures the video signal from the source media and converts it to a digital signal which is then compressed. Video compression removes redundant information between frames and reduces the amount of data necessary to produce a usable image. Compressed videos can be easily transported over networks and require less disk space for storage. Because digital video is stored on a computer, it does not degrade from repeated use or aging, and can be made available concurrently to many different people.

OVS supports **MPEG (Motion Pictures Experts Group)** and **OSF (Oracle Streaming Format)** media formats. To capture and encode video, you have these options:

- Employ a Video Production Studio
- Purchase a Capture/Encoding Station
- Real-time Feed

> **Note:** Real-time feeds and real-time encoding are not covered in this guide. For more information about real-time feeds, refer to the *Oracle Video Server Administrator's Guide and Command Reference.*

Whether you do the video encoding yourself or hire a video production studio, first gather some basic information, as described later in this chapter. **Do not assume that the video production studio will know how you want your files encoded.**

Employ a Video Production Studio

Before investing in an encoding system or purchasing any capture/encoding cards, consider the amount of video encoding you plan to do. It may be more practical and affordable to send your video to a video production studio that specializes in video capture and encoding. When choosing a production studio, research the following:

- price
- skill/experience of the people doing the encoding
- source format requirements
- type of encoders they use
- file output types
- media output (CD-ROM, FTP, tape) provided by the production studio

Purchase a Capture/Encoding Station

This option entails the highest capital cost, but if you plan to encode many hours of high-quality video, it might ultimately be the most cost-effective. Depending on how robust a system you need and how much equipment you already own, the price for setting up your own video capture studio can vary greatly. A typical encoding system requires the following equipment:

Hardware

- Pentium class PC (or higher)
- at least 32 MB of memory
- high-resolution 20-inch monitor
- video display card with 4 MB of VRAM (to get True Color Video on the PC)
- capture card
  - When evaluating capture cards, find the card that drops the least number of frames during capture.
  - Oracle recommends using a card that can capture Motion JPEG or uncompressed / raw format.
  - When choosing a capture card, make sure it is compatible with your PC and has the same video connectors as your video camera.
- necessary disk space (video files can be very large)

Software

- capture and encoding software
- video editing and special effects software (optional)
- codec (compressor / decompressor) that meets your streaming bit rate requirement. A codec is a piece of software that encodes and decodes movie data, usually in a highly compressed format.

Example To support a bit rate of 28.8 kbps, you might choose the Iterated Systems ClearVideo (fractal) codec for video and Voxware AC-10 codec for audio.

Audio/Video Gear

- video tape recorder
- time base corrector
- digital and / or analog video camera
- speakers

> **Note:** For more information about digital and analog video equipment, refer to Chapter 2, "Creating Source Video".

Codecs, Containers, and Frameworks

Digital encoding is not part of the Oracle Video Server and requires third-party encoding hardware and / or software, as described in the previous section. However, simply having digital content on the video server does not mean that all video clients can receive and display the incoming media stream.

Different encoding formats use different **codecs**, **containers**, and **frameworks**:

Codec: a codec (compressor / decompressor) is a piece of software or hardware that encodes and decodes digital media. Each codec uses a different compression format to encode video and / or audio so they require less storage space.

Container: the file format in which an encoder multiplexes, or puts together, video and audio for delivery.

Framework: client software that can understand one of more containers, de-multiplex an incoming network stream, and pass the compressed data to the decoder.

> **Note:** For the purposes of encoding video, you do not need to worry about frameworks. However, if you want more information about frameworks and how they are supported by the OVS system, refer to the *Oracle Video Client Developer's Guide.*

In order for the client to display the video, the video player must be able to intepret and decode the compressed stream.

Codecs (Compressors/Decompressors)

Several compression / decompression formats (codecs) are available for compressing files. The codec you choose affects the visual quality of the movie and the speed with which it plays back on your receiving machine. Different codecs provide different levels of control over compression options.

When choosing a codec, consider:

- Choose a codec that creates a bit rate that remains constant over one-second intervals. Drastic bit-rate fluctuations produce an unpredictable data stream. The Oracle Video Server can stream video better if the bit rate over time is predictable.
- Avoid using the “optimize still” option if your editing software provides it. Content encoded using this option does not stream well via the Oracle Video Server.

After you have encoded your file, verify that your codec encoded the file at the targeted bit rate. Many codecs are not precise and tend to exceed the targeted bit rate.

The Oracle Video Server supports several codecs, including:

- MPEG-1 video
- MPEG-2 video
- MPEG audio
- Iterated Systems ClearVideo
- Intel Indeo
- Radius CinePak
- Voxware MetaSound and MetaVoice

Containers (File Formats)

The Oracle Video Server supports, either directly or indirectly, these container (file) formats:

- MPEG-1 System
- MPEG-2 Transport
- Oracle Streaming Format (OSF)
- Raw Key Frame (RKF)

MPEG-1 System and MPEG-2 Transport

MPEG-1 and MPEG-2 (Motion Picture Experts Group) are standard video formats that provide full screen, full motion (up to 60 frames per second) video with high fidelity / stereo audio playback.

The Oracle Video Server can deliver video using these MPEG codec and container combinations:

- MPEG-1 codecs inside an MPEG-1 System container
- MPEG-1 codecs inside an MPEG-2 Transport container
- MPEG-2 codecs inside an MPEG-2 Transport container

Oracle Streaming Format (OSF)

The Oracle Streaming Format (OSF) is a proprietary stream format developed by Oracle Corporation to enable efficient delivery of media content which has not been optimized for streaming. OSF is an example of a raw key frame container format that is directly supported by OVS.

OSF files are composed of fixed-time blocks, called *records*. There are two types of records, *data* and *init*. Data records include audio, video, and header information. Init records contain setup information, including which audio and video codecs the client needs to decode the file. OSF files must have at least one init record.

Raw Key Frame (RKF)

The Oracle Video Server can support any container format meeting the full raw key frame criteria. To meet the full raw key frame criteria, the encoded file must be both stateless and contiguous.

- **stateless** — all the data for displaying the picture in a video frame is contained entirely in that frame, rather than in any previous frames.

- **contiguous** — all the data for a frame is stored together in the video file without any other data mixed with it.

Audio Visual Interleave (AVI) and Waveform (WAV) files are both examples of container formats that meet this standard. RFK files can be created using any of the codecs listed earlier in this chapter. However, since RKF files like AVI and WAV are not designed for streaming over a network, you must convert them to OSF (Oracle Streaming Format), Oracle's own container format designed for reliable streaming.

Before encoding your source AVI or WAV file, refer to the AVI and WAV encoding guidelines and tips in this chapter. *Make sure that you use **codecs** that your clients support.*

For more information on converting AVI and WAV files to OSF files, refer to "Registering Content Files" in Chapter 4.

How Different Video Clients Support Different Encoding Formats

Although the Oracle Video Server can stream all of the content formats discussed in the previous section, not all formats are supported on all Oracle Video Client machines. Before you encode your content files, be sure to choose an encoding format that your client machine supports.

Table 3–1 lists the types of content that can be currently displayed by the different client machines. This table is subject to change; refer to the *Oracle Video Client Release Notes* for the most recent client support information.

Table 3–1 How Different Video Clients Support Different Encoding Formats

Content Format	PC Client	NC Client	Set-top Box
MPEG-1	✓		✓
MPEG-2			✓
OSF (from AVI and WAV)*	✓	✓	

* For a client to play an incoming OSF file, the appropriate codecs must be installed on the client machine. The NC client supports a specific set of codecs. For a complete list of codecs supported on the NC client, refer to the "Encoding AVI Files for Network Computer (NC) Clients" later in this chapter.

Encoding MPEG Files

This section is divided into the following sections:

- What is MPEG?
- How the Oracle Video Server Supports MPEG-1
- How the Oracle Video Server Supports MPEG-2
- Before You Encode
- MPEG Encoding Samples
- MPEG-1 Encoding Tips
- MPEG-2 Encoding Tips

What is MPEG?

The MPEG (Motion Picture Experts Group) standard addresses the compression and decompression of motion video and audio signals and the synchronization of these signals during playback. MPEG is popular because it provides the following advantages:

- a storage compression ratio of up to 100:1 compared to storing each frame of the video in its original, uncompressed form
- full-motion (up to 30 frames per second), full-screen video with high-fidelity, stereo audio playback

MPEG files are composed of three layers: video, audio, and system (or transport). The video and audio layers (also called elementary streams) contain the coded video and audio data, respectively. The system layer defines the multiplexed structure for the video and audio data as well as the timing information required to replay synchronized segments in real time. Figure 3–1 illustrates the general process of encoding a valid MPEG file.

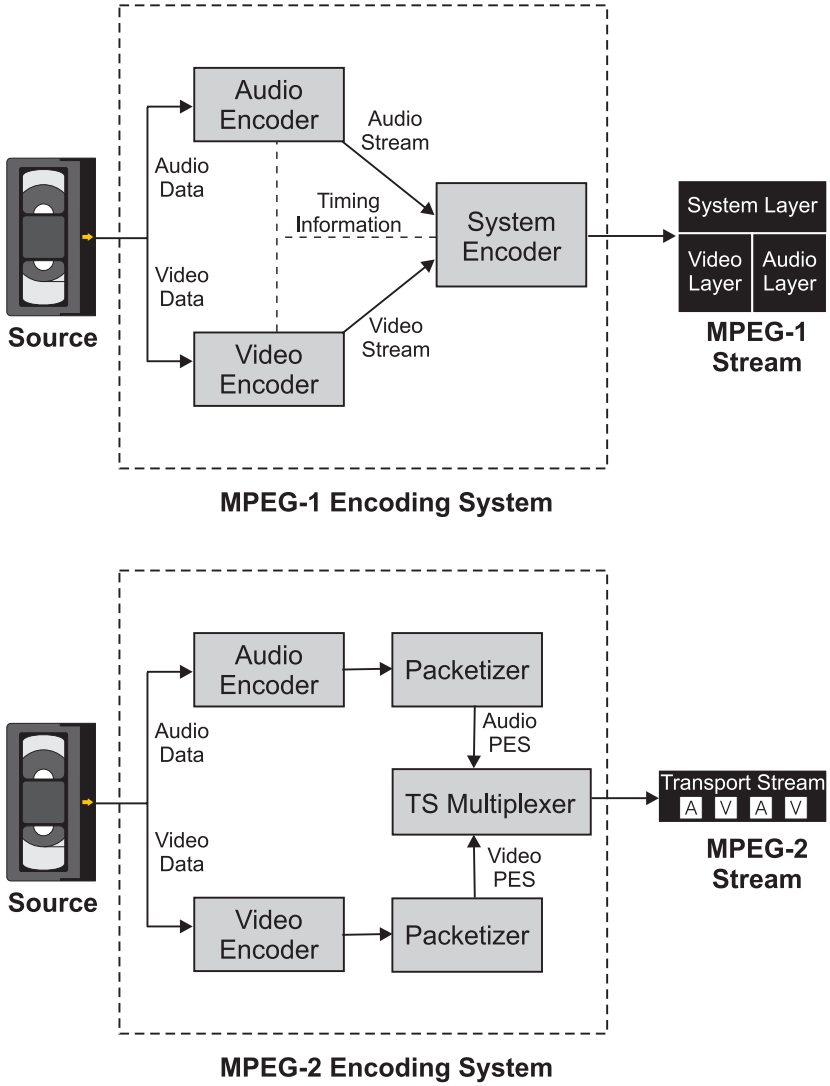
MPEG Standards

There are two widely-used MPEG standards, MPEG-1 and MPEG-2. For a complete technical explanation of these standards, please refer to the appropriate specification (IEC / ISO 11172-1, 2, 3 for MPEG-1 and IEC / ISO 13818-1, 2, 3 for MPEG-2).

Like most standards, the MPEG specifications are written in general terms to support a variety of applications and allow for future innovation. Each encoder / decoder vendor may interpret the specification in subtly different ways. Therefore, to guarantee successful storage and delivery of MPEG files over the Oracle

Video Server system, it is important to understand what the Oracle Video Server requires in an encoded MPEG file.

Figure 3–1 General MPEG Encoding Process



How the Oracle Video Server Supports MPEG-1

Oracle Video Server supports MPEG-1 files encoded at any rate. However, be sure to select an encoding rate that the MPEG decoder on your clients can support. Regardless of the selected encoding rate, the MPEG-1 files must be encoded using:

- MPEG-1 system layer
- MPEG-1 audio layer 2
- MPEG-1 video layer with I-frames

> **Note:** I-frames, or intra-frames, are picture frames compressed using only the redundant data found within the I-frame. In MPEG-1, you can only seek to I-frame boundaries. Therefore, seek requests are only as accurate as the I-frame frequency within a particular file.

For specific encoding recommendations, refer to "MPEG-1 Encoding Tips" later in this chapter.

How the Oracle Video Server Supports MPEG-2

Oracle Video Server supports MPEG-2 files encoded at any rate. However, be sure to select an encoding rate that the MPEG decoder on your clients can support. Regardless of the selected encoding rate, the MPEG-2 files must be encoded using:

- MPEG-2 transport stream (system layer)
- MPEG-1 or MPEG-2 audio layer
- MPEG-1 or MPEG-2 video layer with I-frames

In MPEG-2, system layer coding is specified in two forms: *transport streams* and *program streams*. Each stream is designed for a different set of applications.

> **Note:** For Oracle Video Server applications, program streams are not supported; **you must encode using transport streams**.

Each transport stream must contain only one program. A program consists of compressed elementary streams that have been multiplexed and packetized to produce packetized elementary streams (PES). The Oracle Video Server supports only one video elementary stream per program.

For specific encoding recommendations, refer to "MPEG-2 Encoding Tips" later in this chapter.

Before You Encode

Before encoding your video, gather the information listed in Table 3–2. If you do not own the equipment necessary to encode MPEG files, you will need to provide this information to the video production studio you employ to encode the video. **As a rule, do not assume that the video production studio will know how you want your files encoded.**

Table 3–2 Information to Know Before Encoding MPEG Content Files

Variable	Options
File Format	MPEG-1 system stream. MPEG-2 transport stream Bare elementary streams are not supported
Video Standard	NTSC (U.S. and Japan) or PAL (Europe)
In and Out Points	Start time and end time for each video segment to be encoded
Video Bit Rate	Increments of 400 bps (1,299,200 is a common value for a 1.536 Mbps system rate)
Audio Bit Rate	Increments of 3200 bps (192,000 bps is a common value)
System Padding	MPEG-1: (video bit rate + audio bit rate) x .03 MPEG-2: about 28 kbps
System Bit Rate	video bit rate + audio bit rate + system padding *Common system bit rates:* 2.048 Mbps (for E1) 1.536 Mbps (for T1)
Number of I-Frames per second	Oracle recommends 1 or 2

Table 3–2 Information to Know Before Encoding MPEG Content Files (Cont.)

Variable	Options
Disk Space	Amount of disk space (in megabytes) you need to store the encoded file. To calculate how much disk space an encoded file requires, use this equation: $$disk\ space\ in\ bytes = seconds \times \frac{system\ bit\ rate}{8}$$ *Typical sizes:* 1 hour of 1.366 Mbps video requires about 600 MB of disk space 1 hour of 1.536 Mbps video requires about 700MB of disk space 1 hour of 2.048 Mbps video requires about 900 MB of disk space

When selecting your system bit rate:

- The sum of the bit rate components (audio, video, and padding) *cannot* be larger than the system bit rate.
- Video quality is generally better at a higher encoding rate; however, the encoded file must not exceed the capacity of your transport mechanism (wire) or storage medium (disk).
- Lower encoding rates will allow the server to stream more files at once and save disk space.

For a detailed discussion on planning and configuration issues related to the Oracle Video Server, refer to the *Oracle Video Server Administrator's Guide and Command Reference*.

The information in Table 3–2 will usually be sufficient for the video encoder. If you require more precise control over the MPEG file, discuss it with your encoding vendor.

Encode a Test File First

Not all encoder / decoder vendors handle MPEG files in exactly the same way. Before encoding and storing all your videos, create a test file and play it from a local file system. When you find an encoding / decoding solution that works, encode the rest of your content the same way. As long as the content is MPEG-compliant, the Oracle Video Server will deliver it successfully for normal playback.

MPEG Encoding Samples

This section provides examples of how you might encode video content for delivery over an intranet. The bit rates listed in the examples in Table 3–3 are suggested, but not required.

To view content encoded at the rates listed in Table 3–3, refer to the HTML sample pages located on the Oracle Video Client distribution CD, or click the appropriate link in the table.

Note: To link to and play the sample content, you must have the Oracle Video Client (with the sample applications) and the appropriate codecs installed on the machine from which you are reading this document.

Table 3–3 MPEG Encoding Samples

Encoding Values	Encoding Format and Target Bit Rate			
	MPEG-1 at 1.536 Mbps	**MPEG-1 at 2.048 Mbps**	**MPEG-2 at 4.096 Mbps**	**MPEG-2 at 6 Mbps**
System Bit Rate (bps)	1,536,000	2,048,000	NA	NA
Transport Bit Rate (bps)	—	—	4,096,000	6,144,000
Video Bit Rate (bps)	1,244,000	1,756,000	3,704,000	5,704,000
Audio Bit Rate (bps)	192,000	192,000	192,000	192,000
Audio Sample Rate	44,100 Hz	44,100 Hz	48,000 Hz	48,000 Hz

Note: The combined video and audio bit rates are less than the system bit rate. The remaining bandwidth is used for system information and padding.

MPEG-1 Encoding Tips

This section contains specific encoding tips that advanced encoders should be aware of before encoding MPEG-1 files for use with the Oracle Video Server.

Elementary Streams

The Oracle Video Server does not support the delivery of elementary streams. You must generate a valid MPEG-1 system layer or MPEG-2 transport stream content file.

I-Frame Regularity

The Oracle Video Server expects I-frames to be provided at regular intervals throughout the program stream. Since there is not enough bandwidth in the wire to send every frame as an I-frame, features like rate control (fast-forward, rewind, or seek) can be only as accurate as the interval between I-frames. **Oracle recommends encoding 1 or 2 I-frames per second.**

Note: Some MPEG encoders allow you to define I-frame frequency with the *Group of Pictures (GOP)* variable. GOP size defines the number of picture frames for every I-frame. A smaller GOP will provide better response when searching through the content (I-frames are closer together), but does not provide as high a compression ratio. **A GOP size of 15 provides two I-frames per second.**

Non-sequential PTS and SCR

The MPEG specification permits Presentation Time Stamp (PTS) and System Clock Reference (SCR) data to be streamed out of sequence so long as there is a discontinuity indication in the encoded file. However, some clients query the clock from the decoder chip to identify locations within a particular file. If content is encoded with time stamps that occur out of numerical sequence or multiple times, the times become meaningless, and the decoder will have difficulty presenting the file. **Oracle strongly recommends encoding PTSs and SCRs so that their values increase in a non-repeating, sequential pattern.**

Sequence Header Manipulation

The MPEG-1 specification permits encoders to change the sequence header information mid-program. The sequence header contains matrices that provide critical timing, frame content, color-map, and decoding information to the client. If these matrices

change in a piece of content, search functions will not work properly. **Oracle recommends that you do not change the sequence header in mid-program.**

MPEG-2 Encoding Tips

This section contains specific encoding tips that advanced encoders should be aware of before encoding MPEG-2 files for use with the Oracle Video Server.

Elementary Streams

The Oracle Video Server does not support the delivery of elementary streams. You must generate a valid MPEG-2 transport stream content file.

MPEG-1 or MPEG-2 Video Layer

MPEG-2 transport streams can contain an MPEG-1 video layer that adheres to the MPEG-1 encoding guidelines discussed earlier in this chapter.

Non-sequential PCR

The MPEG specification permits Program Clock Reference (PCR) data to be streamed out of sequence so long as there is a discontinuity indication in the encoded file. However, some clients query the clock from the decoder chip to identify locations within a particular file. If content is encoded with time stamps that occur out of numerical sequence or multiple times, the times become meaningless, and the decoder will have difficulty presenting the file. **Oracle strongly recommends encoding PCRs so that their values increase in a non-repeating, sequential pattern.**

NPT Descriptors

The DSM-CC specification recommends encoding NPT (Normalized Play Time) descriptors in the PMT (Program Map Table). **This encoding method does not work. Do not encode NPT descriptors in your PMTs.**

The Oracle Video Server indicates command transitions by inserting NPT descriptors automatically. The **commandID** in this descriptor is incremented with each command, and the NPT in the descriptor will indicate the actual presentation time of the delivered content.

PAT and PMT Manipulation

The MPEG-2 specification permits encoders to change the Program Association Table (PAT) and Program Map Table (PMT) information mid-program. These tables point to critical video and audio data necessary to decode the content successfully. **Oracle strongly recommends that you do not change the PATs or PMTs mid-program**.

PCR Placement

In MPEG-2, you are able to place the Program Clock Reference (PCR) on any Program ID (PID). Some VCR operations in the Oracle Video Server work best when the PCR is on the same PID as the video. **Oracle recommends encoding the PCR on the same PID as the video.**

Sequence Header and I-Frame Regularity

During rate control operations, decoders depend on sequence headers and I-frames to produce quick and accurate playback. Oracle encourages encoding one sequence header per I-frame.

Sequence Header Manipulation

The MPEG-2 specification permits encoders to change the sequence header information mid-program. The sequence header contains matrices that provide critical timing, frame content, color-map, and decoding information to the client. If these matrices change in a piece of content, random access becomes problematic. **Oracle strongly recommends that you do not change the sequence header mid-program.**

Program Specification Information

Transport streams that contain multiple programs often include Program Specification Information (PSI), such as CATs and NITs. Since the Oracle Video Server supports only single-program transport streams, **do not encode PSIs into your transport stream.**

Encoding AVI Files

This section is divided into the following sections:

- What is AVI?
- How the Oracle Video Server Supports AVI
- Before You Encode
- AVI Encoding Samples
- AVI Encoding Tips

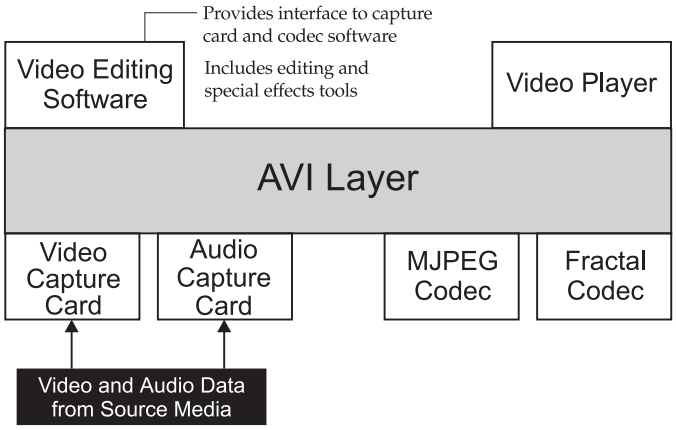
What is AVI?

The Microsoft audio-video interleaved (AVI) file format is a RIFF (Resource Interchange File Format) file specification used with applications to capture, edit, and play back audio-video sequences. Within an AVI file, which typically includes both an audio and video sequence, the audio and video data can be generated by any compression method, such as Intel Indeo, Iterated Systems Clearvideo (fractal), and Motion JPEG. To play back the compressed AVI file, the receiving machine must have a corresponding codec (compressor-decompressor).

The compression method you choose will depend upon the hardware and software on the machines in your environment and the transport medium you will be streaming video over (low-speed or high-speed modem, LAN, WAN).

Figure 3–2 illustrates how the AVI capture, editing, and encoding components work together in your client machine.

Figure 3–2 How the AVI Components Fit Together



How the Oracle Video Server Supports AVI

AVI files are *not* designed for streaming over a network. Therefore, after you encode your AVI file, you must convert it to an OSF (Oracle Streaming Format) file, which is optimized for streaming. To learn how to convert an AVI file to OSF, refer to "Registering Content Files" in Chapter 4.

To ensure that your AVI file will stream properly after it is converted to OSF, you must encode your AVI file in one of the following ways:

- one audio sequence and one video sequence
- video only
- audio only

Oracle Video Server also requires that the AVI file contain a valid index chunk. The index chunk specifies the location of data chunks within the file, allowing the client application to locate a specific point in the video quickly.

For more information on the AVI file structure, refer to Microsoft's *Win32:SDK Developer's Guide.*

Encoding AVI Files for Network Computer (NC) Clients

If you are creating content that you are intending to stream to a Network Computer client, make sure that you encode the content using formats that the NC client supports. Currently, these formats include:

- Iterated Systems ClearVideo 1.1 (video only)

  For more information on using the Iterated Systems ClearVideo codec (including procedures and tips for editing and compressing video with Adobe Premiere), visit the Iterated Systems web site at **http://www.iterated.com**.

- Voxware MetaSound AC-8 (audio only)
- PCM (pulse code modulation)

  PCM is an uncompressed audio encoding format. On NCs, OVS supports PCM audio content which has been encoded using the 16-bit and mono capture settings. You can encode PCM at 8K, 11K, 22K, or 44K samples per second. When using the 16-bit and mono settings, these sample rates equate to 128kbps, 176kbps, 352kbps, and 704kbps audio signals, respectively.

Note: For the most recent list of encoding formats supported by the Network Computer, refer to the *Oracle Video Server Release Notes*.

Before You Encode

Before encoding your video, gather the information listed in Table 3–4. If you do not own the equipment necessary to encode AVI files, you will need to provide this information to the video production studio you employ to encode the video. **As a rule, do not assume that the video production studio will know how you want your files encoded.**

Table 3–4 Information to Know Before Encoding AVI Content Files

Variable	Options
File Format	AVI
Codec(s)	Full Raw Key Frame (Select a codec that your client supports)
Video Standard	NTSC (U.S. and Japan) or PAL (Europe)
In and Out Points	Start time and end time for each video segment to be encoded
Frame Rate	For low bit rate (28k or less) video, Oracle recommends using 6 to 8 fps For video delivered at bits rate higher than 28k, Oracle recommends using 12 fps or above
Resolution	Common settings: 160 x 120 (QSIF) 176 x 144 (QCIF) 320 x 240 (SIF) 352 x 288 (CIF)
Video Bit Rate	Measured in bits per second
Audio Bit Rate	Measured in bits per second
Overhead	(video bit rate + audio bit rate) x .03
Total Bit Rate	video bit rate + audio bit rate + overhead
Key Frame frequency	Oracle recommends at least 1 key frame every 3 seconds.
Target Bit Rate	To calculate the bit rate for a particular file, use this equation: $bitrate = \frac{file\ size\ in\ bytes \times 8}{seconds}$
Disk Space	Amount of disk space (in megabytes) you need to store the encoded file. To calculate how much disk space an encoded file requires, use this equation: $disk\ space\ in\ bytes = seconds \times \frac{system\ bit\ rate}{8}$ *Typical sizes:* 1 hour of 28K video requires about 12 MB of disk space 1 hour of 112K video requires about 50 MB of disk space 1 hour of 1.366 Mbps video requires about 600 MB of disk space 1 hour of 1.536 Mbps video requires about 700MB of disk space 1 hour of 2.048 Mbps video requires about 900 MB of disk space

AVI Encoding Samples

This section provides examples of how you might encode AVI files that you plan to convert to OSF files. The bit rates listed in the examples in Table 3–5 are suggested, but not required.

Using these codecs, you can create low bit rate files which use the AVI container. Then, you can use Oracle Video Server Manager or the **vsmkosf** utility to convert AVI files into OSF files, as described in Chapter 4.

To view content encoded at the rates listed in Table 3–5, refer to the HTML sample pages located on the Oracle Video Client distribution CD, or click the appropriate link in the table.

> **Note:** To link to and play the sample content, you must have the Oracle Video Client (with the sample applications) and the appropriate codecs installed on the machine from which you are reading this document.

Table 3–5 AVI Encoding Samples

Encoding Values	Encoding Format and Target Bit Rate			
	AVI at 28 kbps	**AVI at 128 kbps**	**AVI at 450 kbps**	**AVI at 800 kbps (PC client only)**
Video Format*	Iterated Systems ClearVideo	Iterated Systems ClearVideo	Iterated Systems ClearVideo	Indeo Video Interactive
Audio Format*	Voxware AC-10	Voxware AC-10	PCM	uncompressed 16-bit mono, 11kHz
Frame Rate	6 fps	12 fps	15 fps	15 fps
Resolution	160 x 120	320 x 240	320 x 240	320 x 240
Video Bit Rate	12 kbps	94 kbps	256 kbps	1000 kbps
Audio Bit Rate	10 kbps	10 kbps	176 kbps	176 kbps

Note: The combined video and audio bit rates are less than the target bit rate. The remaining bandwidth is used for overhead.

* Different editing software packages do not support encoding formats in the same way. You will need to experiment with these values with your specific editing software to produce video that meets your needs.

AVI Encoding Tips

This section contains encoding tips that encoders should be aware of before creating files for use with the Oracle Video Server.

Encoding Rate

- In general, higher encoding rates produce better results. However, be sure to select a rate that does not exceed the playback rate of the computer system on which you intend to play the video and the bandwidth available on the network. Selecting too high of a rate could result in dropped frames and flutter during playback.
- Some encoding methods, such as Motion JPEG, are not appropriate for low-bandwidth connections because every frame must be a key frame.
- When choosing your encoding rate, allow about 3% of overhead.
- If you plan to deliver video over an ISDN line, assume that the ISDN line has only 112K of throughput.
- If you plan to deliver video over a 28.8 kbps modem, target a combined audio/video bit rate of 22K.
- Some encoding formats, like Intel Indeo, do not allow you to set a specific video encoding rate. Instead, you must define an encoding rate that does not vary too much from your target total bit rate. In most editing/encoding software packages, this can be accomplished by:
  - Setting the quality bar to 100%
  - Setting the Limit Data Rate option to your target total bit rate
  - Enabling the CD-ROM option

Key frames

Key frames are the base frames against which other frames are compared for differences. Key frames are saved in their entirety, while intervening frames are compressed based on their differences from their adjacent frames (often called delta frames).

Since key frames contain a self-defined image, the Oracle Video Server uses them to support rate control features like scan and seek. For low-bandwidth connections, there is not enough room in the wire to send all key frames. Therefore, you must choose a compression format that allows you control key frame frequency. Rate control can be

only as accurate as the interval between key frames. **Oracle recommends encoding at least one key frame every three seconds.**

> **Example:** If the playback rate of your video is 15 frames per second, at least one out of every 45 consecutive frames should be a key frame.

Encoding WAV Files

The Microsoft waveform-audio file (WAV) is a format for recording digital audio files. Most sound cards and encoders are able to save audio in WAV format. For specific guidelines about recording WAV files with your particular card, refer to the documentation provided with your card.

WAV files are *not* designed for streaming over a network. Therefore, after you encode your WAV file, you must use Oracle Video Server Manager or the **vsmkosf** utility to convert it to an OSF file, which is optimized for streaming.

The Oracle Video Server can handle a WAV file of any size. However, be sure that the file size does not exceed the capacity of your transport medium, as described in the previous sections. File size is dependent on the sampling rate at which you record and the length of the audio clip.

Examples

This section provides examples of how you might encode WAV files that you plan to convert to OSF. The bit rates listed in the examples in *Table 3–6* are suggested, but not required.

To view content encoded at the rates listed in Table 3–6, refer to the HTML sample pages located on the Oracle Video Client distribution CD, or click the appropriate link in the table.

> **Note:** To link to and play the sample content, you must have the Oracle Video Client (with the sample applications) and the appropriate codecs installed on the machine from which you are reading this document.

Table 3–6 WAV Encoding Samples

Encoding Values	Encoding Format and Target Bit Rate		
Audio Format	**WAV at 28k (modem rate)**	**WAV at 112k (ISDN rate)**	**WAV at 112k (ISDN rate)**
Sampling Rate	Voxware AC-16	IMA-ADPCM	IMA-ADPCM
Audio Quality	16-bit, mono	22 kHz	11 kHz
Audio Bit Rate	16 kbps	4-bit, mono	4-bit, stereo

Note: The audio bit rate is less than the target bit rate. The remaining bandwidth is used for overhead.

4

Loading and Managing Digital Content

After you have encoded your content files, as described in Chapter 3, you must load and register your content. Once registered, the content files can be managed and manipulated in several ways with the Oracle Video Server system.

Topics covered in this chapter are:

- Content Administration Tasks
- Loading Content Files
- Copying Content Files
- Archiving Content Files
- Deleting Content Files
- Verifying the Integrity of an MPEG-2 Transport File
- Registering Content Files
- Managing Registered Content

Content Administration Tasks

The Oracle Video Server system provides two content management interfaces:

- Oracle Video Server Manager console, a graphical, Java-based application
- the OVS command-line utilities

For a brief overview on the benefits of each of these interfaces, refer to the “Content Management Interfaces” section in Chapter 1.

Table 4–1 lists the types of tasks that you can perform from each of these interfaces. For tasks performed from the command line, a full description and step-by-step instructions are provided in this document. For tasks performed from VSM, a brief description is provided in this document, and then you are referred to the appropriate Video Server Manager online Help topic, if you require more information.

Table 4–1 Content Management Tasks Available from VSM and the OVS Command Line

Task	Video Server Manager	OVS Command Line
Load content into the Oracle Media Data Store (MDS)	✓	✓
Copy content between MDS volumes	✓	✓
Register content with the MDS and Oracle database	✓	✓
View the contents of logical content and clips	✓	
Create program schedules for broadcast applications	✓	
Create logical content in the Oracle database	✓	✓
Edit logical content and clips stored in the Oracle database	✓	
Delete logical content stored in the Oracle database	✓	✓
Delete content from the Oracle MDS	✓	✓
Modify tag files		✓
Display the contents of tag files		✓
Archive content		✓
Verify the Integrity of an MPEG-2 transport file		✓

Note: For more detailed information about how to manage content from Video Server Manager, refer to the Video Server Manager online Help.

Loading Content Files

You can load content into the MDS using a local CD-ROM or tape drive, or using remote file access over a network. The OVS supports online loading. That is, you can load content while the server is providing uninterrupted media streams to clients. The OVS software also guarantees that content loading will not interfere with the delivery of real-time feeds.

Note: Before loading content files into an MDS volume for the first time, you must initialize the volume using the **mdsvolinit** utility. Refer to Chapter 7 "Oracle Media Data Store Tasks and Procedures" of the *Oracle Video Server Administrator's Guide and Command Reference* for information on **mdsvolinit**. Also, before initializing the volume, make any needed modifications to your MDS volume or the **voltab** file, such as:

- adding disks
- changing the size of the RAID sets
- enabling or disabling RAID protection
- changing the stripe width

Once a volume contains content files, you cannot perform any of the above operations; doing so will destroy all content. However, you can always create new volumes should you require additional storage capacity.

To load content files into the Oracle MDS, you can:

- use the Load Content utility from VSM
- use one of these OVS command-line utilities:
  - **mdstar –** to load files from a tar archive
  - **mdscopy –** to load files from a CD-ROM or host file system
  - **ftp –** to load files across a network

Note: For more information about the conventions used for naming MDS volumes and files, refer to "Naming MDS Volumes and Files" in Appendix A.

Loading Content with Video Server Manager

The VSM Load Content utility automates the tasks associated with loading, converting, and registering content files to be used with the Oracle Video Server.

The Load Content utility performs these tasks:

- copies the content file(s) from the source media to an MDS volume
- converts raw key frame (RKF) files, such as AVI and WAV, into Oracle Streaming Format (OSF) files
- registers the MPEG or OSF file(s) with the MDS and database

> **Note:** To save space in the MDS, the source RKF file will be removed from the MDS after the file is converted to OSF.

To load content with the VSM Load Content utility:

1. Verify the path and file name for files stored in a remote file system or on the CD-ROM, or the name of the tape device.
2. Click the **Load Content** button in the VSM toolbar.
3. Fill in the appropriate fields and click **OK**.

> **Note:** The VSM Load Content utililty can also extract content files from a tar archive that resides on a single tape.

For more detailed information about this procedure, refer to the “Loading and Registering Content” topic in the VSM online Help.

Loading Content Using mdstar

To load an encoded content file that resides in a tar archive, use the **mdstar** command-line utility. The **mdstar** utility enables you to create or extract a tar archive between the MDS and a host file system or tape device.

> **Note:** Verify the block size of your tar archive. The default block size of the Oracle MDS may differ from your OS tar. The default block size for **mdstar** is 128.

Example 4–1 Loading Content from Tape to the MDS Using mdstar

This example extracts all files with the extension **.mpg** from tape device **/dev/scsi/rmt7** and places them in the MDS volume **video**:

```
% mdstar -x -b 128 -f /dev/scsi/rmt7 -p /mds/video "*.mpg"
```

where:

-b specifies the blocking factor to be used. In this example the blocking factor specified is 128; **mdstar** defaults to 20.

-f is the name of the device that contains the archive

-p is the volume in which the extracted files will be placed

-x extracts a file from a tar archive to the specified MDS volume

Loading Content Using mdscopy

To load encoded video that resides on a CD-ROM, removable disk drive, or host file system, use the **mdscopy** command-line utility. The **mdscopy** utility enables you to copy files from one MDS volume to another, or from a host file system to the MDS.

Example 4–2 Loading a File from a Host File System to an MDS Volume Using mdscopy

This example copies the file **oracle1.mpg** from the current directory on a host file system to the MDS volume **video**:

```
% mdscopy ./oracle1.mpg /mds/video
```

Note: The **mdscopy** utility requires that the full MDS path of the output file be explicitly stated, or that the **MDS_CWD** environment variable be set to the appropriate MDS volume. For information on **MDS_CWD,** refer to *Setting the MDS_CWD Environment Variable* section in Chapter 7 of the *Oracle Video Server Administrator's Guide and Command Reference*.

Example 4–3 Loading Multiple Files from a Host File System to an MDS Volume Using mdscopy

This example uses a wildcard search to copy all **.mpg** files from the current host directory into the MDS volume **video**:

UNIX
```
% mdscopy ./*.mpg /mds/video
```

WIndows NT
```
for %f in (.\*.mpg) do mdscopy %f /mds/video
```

> **Note:** On Windows NT, the command shell does not expand wildcards by default. Therefore, you must use the syntax shown above to load multiple files.

Loading Content Using FTP

The OVS supports most operating system and network vendor FTP clients. Using an FTP client, you can transfer binary files across a network from a host file system to an MDS volume.

> **Note:** In order to FTP files to or from an MDS volume, the MDS FTP server (**mdsftpsrv**) must be running. For more information on MDS FTP refer to the section *Enabling FTP Access to the MDS* in the *Oracle Video Server Administrator's Guide and Command Reference.*

Like most FTP servers, the MDS FTP server requires you to log in with a user name and a password. Authentication of this information depends on the operating system.

Example 4–4 Loading Content From a UNIX Host to the MDS Using FTP

This example transfers the file **oracle2.mpg** from a UNIX host to the MDS.

1. Establish the FTP connection to the OVS server machine by specifying the appropriate server name and port number:

```
% ftp oracle-sun 1621
Connected to oracle-sun.us.oracle.com.
220 Oracle Media Data Store FTP Server ready.
Name (oracle-sun:aloyusious):
331 Password required for user aloyusious.
Password:
230 Welcome to MDS server.
```

Because the MDS FTP server co-exists with standard FTP servers, it must listen on a different port. For information on the MDS FTP server, and the specified port number, refer to the *Enabling FTP Access to the MDS* section in Chapter 7 of the *Oracle Video Server Administrator's Guide and Command Reference.*

2. Using the **cd** command, change to the volume from which you want to retrieve data. This example changes to **video2**:

```
ftp> cd video2
250 CWD command successful.
ftp> pwd
257 /mds/video2 is current volume.
```

3. Set the file transfer type to **binary**:

```
ftp> binary
200 Type set to BINARY.
```

4. When transferring a file from a host file system to the MDS, you must specify the **allo** command. The **allo** command creates a file of the specified size on the MDS file system. This example enables you to transfer any file up to 24 MB (24,000,000 bytes):

```
ftp> quote allo 24000000
200 ALLO command successful.
```

   For more information on **allo** and its usage on various FTP clients, refer to the section "Sending the allo Command" in Appendix A.

5. Issue the **put** command to transfer a file from a host file system to the MDS:

```
ftp> put oracle2.mpg
200 PORT command successful.
```

   This example transfers the file **oracle2.mpg** to the working MDS volume (**video2**).

For more information on FTP, refer to the section "FTP Usage Notes" in Appendix A or refer to the FTP documentation provided with your server.

Copying Content Files

The Oracle Video Server system provides several utilities to copy files between MDS volumes. As with loading content, you can transfer content between volumes while the server is providing uninterrupted media streams to clients. The OVS software guarantees that content transferring will not interfere with the delivery of real-time feeds.

To copy content files between MDS volumes, you can:

- use the Load Content utility from VSM
- use the **mdscopy** utility from the Oracle Video Server command line
- use passive FTP to copy content between two video servers

> **Note:** For more information about the conventions used for naming MDS volumes and files, refer to the section "Naming MDS Volumes and Files" in Appendix A.

Copying Content with Video Server Manager

To copy content between MDS volumes with the VSM Load Content utility:

1. Verify the path and file name for file(s) that you want to copy.
2. Click the **Load Content** button in the VSM toolbar.
3. Fill in the appropriate fields and click **OK**.

For more detailed information about this procedure, refer to the "Loading and Registering Content" topic in the VSM online Help.

Copying Content Using mdscopy

To copy content between MDS volumes using the OVS command line, use the **mdscopy** utility.

Example 4–5 Copying a File from one MDS Volume to Another

This example copies the file **oracle1.mpg** from one MDS volume (**volume1)** to another (**volume2)**:

```
% mdscopy /mds/volume1/oracle1.mpg /mds/volume2/oracle1.mpg
```

For further information on **mdscopy**, refer to the *Oracle Video Server Administrator's Guide and Command Reference.*

Copying Content Using Passive FTP

From time to time you might need to transfer files from one video server to another. Using passive FTP, you can copy files directly between the two servers. Passive FTP provides these advantages:

- you can copy content between servers from a remote machine
- you can efficiently copy large amounts of data without having to worry about running out of space on your local hard drive or tape device
- you save time by copying the files only one time

Example 4–6 Copying a File Between Two Video Servers Using Passive FTP

This example copies the MPEG file **movie.mpg** from the volume **video** on video server **OVS2** to the volume **video** on video server **OVS1**.

> **Note:** In this example, the volumes name (**video**) on **OVS1** and **OVS2** is identical. If you FTP registered content to a volume of a different name, you must run the **vstagpatch** utility to ensure that the OVS MDS and database registry point to the correct file. Furthermore, even if the volumes have the same name, you will need to update the database registry (if available), as described later in this chapter.

1. Establish the FTP connection to **OVS1** and **OVS2** by specifying the server name and port number:

   ```
   %ftp OVS1 1621
   ftp> proxy open OVS2 1621
   ```

2. Using the **cd** command, change to the volume on **OVS1** from which you want to copy the content file:

   ```
   ftp> cd video
   ```

3. Using the **cd** command, change to the volume on OVS2 to which you want to copy the content file:

   ```
   ftp> proxy cd video
   ```

4. Set the file transfer type to **binary** for both video servers:

```
ftp> bin
ftp> proxy bin
```

5. You must specify the **allo** command. The **allo** command creates a file of the specified size on the MDS file system. This example enables you to transfer any file up to 24 MB (24,000,000 bytes):

```
ftp> quote allo 24000000
```

   For more information on **allo** and its usage on various FTP clients, refer to the section "Sending the allo Command" in Appendix A.

6. FTP the file **movie.mpg** from **OVS2** to **OVS1**.

```
ftp> proxy put movie.mpg
```

Archiving Content Files

From time to time, you might choose to archive content files which have not been requested for a long period of time to make room for other content files.

The Oracle Video Server system provides two utilities to archive MDS content files. As with loading and transferring content, you can archive content files while the server is providing uninterrupted media streams to clients. The OVS software guarantees that content archiving will not interfere with the delivery of real-time feeds.

To archive MDS content files, you can use the following OVS command-line utilities:

- **mdstar**
- **ftp**

> **Note:** You can also use the **mdshsmctl** utility to archive content onto a tape of CD stored in a Hierarchical Storage Management (HSM) system. For more information about archiving content into an HSM tertiary storage device, refer to the *Oracle Video Server Administrator's Guide and Command Reference*.

Archiving MDS Content Files Using mdstar

To archive content files from the MDS to a tertiary tape device, or into a single tar file, use the **mdstar** utility as described in the following examples.

Example 4–7 Archiving Content from the MDS to Tape Using mdstar

This example archives the file **oracle1.mpg** in the MDS volume **video** to the device **/dev/scsi/rmt8**. The **-c** option specifies that **mdstar** create an archive in the location specified with **-f**:

```
% mdstar -c -T -f /dev/scsi/rmt7 /mds/video/oracle1.mpg
a oracle1.mpg, 1610612789 bytes, 1 segment(s), 293 kb/s
```

This example uses the default blocking factor of 20. To ensure that you are using the correct blocking factor, refer to the documentation that came with your server machine.

Note: This example and others in this section use the **-T** option. The **-T** option enables verbose mode, which lists additional information about actions executed on files.

Example 4–8 Archiving All Files in an MDS Volume into a Single File

This example uses a wildcard search to archive all files in the MDS volume **video** with the extension **.mpg** to the file **film.tar**:

```
% mdstar -c -T -f /tmp/film.tar "/mds/video/*.mpg"
a oracle1.mpg, 1610612789 bytes, 1 segment(s), 278 kb/s
a oracle2.mpg, 3610612 bytes, 1 segment(s), 289 kb/s
```

For further information on **mdstar**, refer to the *Oracle Video Server Administrator's Guide and Command Reference.*

Archiving MDS Content Files Using FTP

The OVS supports most operating system and network vendor FTP clients. Using an FTP client, you can transfer binary files across a network from a host file system to an MDS volume.

Note: In order to FTP files to or from an MDS volume, the MDS FTP server (**mdsftpsrv**) must be running. For more information on MDS FTP refer to the section *Enabling FTP Access to the MDS* in Chapter 7 of the *Oracle Video Server Administrator's Guide and Command Reference.*

Like standard FTP servers, the MDS FTP server requires you to log in with a user name and a password. The authentication of this information depends on the operating system.

Example 4–9 Archiving MDS Content Files Using FTP

This example transfers the file **oracle1.mpg** from the MDS to a UNIX host.

1. Establish the FTP connection to the OVS server machine by specifying the appropriate server name and port number:

```
% ftp oracle-sun 1621
Connected to oracle-sun.us.oracle.com.
220 Oracle Media Data Store FTP Server ready.
Name (oracle-sun:aloyusious):
331 Password required for user aloyusious.
Password:
230 Welcome to MDS server.
```

   Because the MDS FTP server co-exists with standard FTP servers, it must listen on a different port. For information on the MDS FTP server, and the specified port number, refer to the *Enabling FTP Access to the MDS* in Chapter 7 of the *Oracle Video Server Administrator's Guide and Command Reference.*

2. Issue an **ls** command to list the available MDS volumes:

```
ftp> ls
200 PORT command successful.
150 Data connection for MDS directory
video
video2
226 Transfer completed.
15 bytes received in 0.0014 seconds (10 Kbytes/s)
```

   This example indicates that two volumes (**video** and **video2**) are available.

3. Using the **cd** command, change to the volume from which you want to retrieve data. This example changes to **video2**:

```
ftp> cd video2
250 CWD command successful.
ftp> pwd
257 /mds/video2 is current volume.
```

4. Before transferring the file, you must use the **binary** command to put the client into binary mode. By default, most FTP clients start up in ASCII mode. However, the MDS FTP server contains binary files (digitized audio and video files) and therefore supports only binary file transfers. Set the file transfer type to **binary**:

```
ftp> binary
200 Type set to BINARY.
```

5. Issue the **get** command to transfer a file from the MDS to a host file system. In this example, the file **oracle1.mpg** is transferred to the working directory on the host:

```
ftp> get oracle1.mpg
200 PORT command successful.
150 Data connection for oracle1.mpg
226 Transfer completed.
local: oracle1.mpg remote: oracle1.mpg
14854692 bytes received in 22 seconds (6.5e+02 Kbytes/s)
```

Deleting Content Files

The Oracle Video Server system provides the **mdsdelete** command-line utility to delete files from MDS volumes.

> **Note:** To protect your content files being inadvertantly deleted when using the VSM console, VSM prevents you from deleting content files directly from the MDS. You can delete MDS content from VSM only via the delete logical content, clips, and physical content procedures described later in this chapter.

Deleting MDS Content Files Using mdsdelete

To delete content files from the MDS, use the **mdsdelete** utility as described in the following example.

Example 4–10 Deleting a Content File from the MDS Using mdsdelete

This example deletes the file **oracle1.mpg** from the MDS:

```
% mdsdelete oracle1.mpg
```

When the delete operation is complete, the associated disk space will be available for new content.

> **Note:** If you using the OVS with a database, verify that your content files can be deleted safely. The database might contain logical content, clips, and physical content that require the content file you plan to delete.

Verifying the Integrity of an MPEG-2 Transport File

Before registering an MPEG-2 transport file with VSM or an OVS command-line utility (which can be a lengthy process), you can check the integrity of your file with the **vsmpegchk** utility. The **vsmpegchk** utility will returns a message indicating whether the file is valid and can be registered, or is corrupt.

> **Note:** Although a file might be deemed valid by **vsmpegchk**, the **vstag** still might reject the file. The **vstag** utility performs a more detailed examination of the content file during content registration.

Example 4–11 Verifying the Integrity of a Single MPEG-2 Transport File

This example verifies the files **oracle1.mpg2** in the MDS volume **video**:

```
% vsmpegchk /mds/video/oracle1.mpg2

oracle1.mpg2: ok
```

Example 4–12 Verifying the Integrity of Multiple MPEG-2 Transport Files

This example verifies the integrity of files **oracle1.mpg2** and **oracle2.mpg2** in the volume **video**:

```
% vsmpegchk /mds/video/oracle1.mpg2 /mds/video/oracle2.mpg2

oracle1.mpg2: ok
oracle2.mpg2: error at position 188: 23 45 31
```

The returned message indicates that:

- **oracle1.mpg2** is valid
- **oracle2.mpg2** has invalid MPEG-2 data at address 188, and the data at that location is 23, 45, and 31.

Registering Content Files

Before you can play content files from the Oracle Video Server (OVS), you must register the file with the Oracle MDS and database (if available). For each content file that you register, you generate:

- a tag file which is stored in the same MDS as the media content file.

  Tag files contain important metadata information, such as encoding rate, time length, and data used for rate control operations (seek forward, seek backward, pause) that the client application uses when accessing the video content file. Tag files must be stored in the same MDS as the content file.

- logical content, clip, and physical content objects that are stored persistently in the database.

> **Note:** The OVS content model, described in Chapter 1, assumes that each content file will be registered only once with the Oracle MDS and database. **Although OVS does not prevent you from creating multiple registries, Oracle strongly recommends that you do not register a content file more than once, as it makes content management extremely complex and difficult.**

Depending on the type of content that you want to register, or the type of tag file that you want to create, you register content files one of the following ways:

- use the Register Content utility from VSM
- use one of these command-line utilities:
  - **vstag:** registers an MPEG or OSF file
  - **vsmkosf**: converts an AVI or WAV file into an OSF file and registers the OSF file
  - **vsgentag:** registers an MPEG or raw key frame file. Unlike **vstag**, **vsgentag** creates a *null tag file*, which does not allow rate control operations. Use **vsgentag** to register content where rate control is not essential or proprietary user codecs are used.

Rate Control

Rate control is the ability to play video at different speeds and in different directions, to seek in either direction, to scan in either direction, or to pause and resume playback of video files.

When you register content, you can also prohibit users from using certain rate control operations. Currently, OVS allows you to prohibit only pause, scan forward, and scan backward operations. Other prohibitions that you specify are not implemented.

Registering Content with Video Server Manager

You can use the VSM Register Content utility to register any MPEG or OSF file stored in the Oracle MDS.

Note: If you used the VSM Load Content utility to load your content into the MDS (as described earlier in this chapter), the content file was registered automatically. However, if the content file was loaded into the MDS using one of the command-line utilities, or if the database objects and tag files have been deleted, you can register the content using VSM or one of the command-line utilities.

To register a content file with VSM:

1. Click the **Register Content** button in the VSM toolbar.
2. Fill in the appropriate fields and click **OK**.

For more detailed information about this procedure, refer to the "Loading and Registering Content" topic in the VSM online Help.

Registering Content Using vstag

These example use **vstag** to register MPEG and OSF files.

Example 4–13 Registering an MPEG-1 File with vstag

This example registers the MPEG-1 file **oracle1.mpg** located in the MDS volume **video**. The **-E** option specifies the extension .**mpi** to be used in the tag file name:

```
% vstag -E mpi /mds/video/oracle1.mpg
```

Example 4–14 Registering an MPEG-2 File with vstag

This example tags the MPEG-2 file **oracle1.m2t** located in the MDS volume video. The **-E** option specifies the extension **.mpi** to be used in the tag file name:

```
% vstag -E mpi /mds/video/oracle1.m2t
```

Example 4–15 Registering an OSF File with vstag

This example registers the OSF file **oracle1.osf** located in the MDS volume **video**. The **-E** option specifies the extension .**mpi** be used in the tag file name:

```
% vstag -E mpi /mds/video/oracle1.osf
```

> **Note:** The **vstag** utility requires that the full MDS path of the input file be explicitly stated, or that the **MDS_CWD** environment variable be set to the appropriate MDS volume. For information on **MDS_CWD** refer to the *Setting the MDS_CWD Environment Variable* section in Chapter 5 of the *OVS Administration and Reference Guide*.

Example 4–16 Registering a Corrupted Content File with vstag

Some encoders introduce occasional errors (like truncating a sequence header) when encoding content files. Normally, **vstag** will abort when it encounters such an encoding error. However, by using **vstag** with the **-i** option, you can register content files which contain encoding errors. Often, even if these errors exist in the registered content file, OVS can still successfully stream the file.

This example registers the file **oracle1.mpg** in the volume **video**:

```
% vstag -E mpi -i /mds/video/oracle1.mpg
```

Example 4–17 Registering Multiple MPEG Files with vstag

This example uses a wildcard search to register all files with the extension **.mpg** in the MDS volume **video**. Note the use of quotes on the volume name. When using a wildcard search on an MDS volume, the full volume name must be specified in the search string:

```
% vstag -E mpi "/mds/video/*.mpg"
```

Example 4–18 Loading and registering a Content File in One Command

Using the **-o** option, you can both copy and register a content file in one step. This example registers the file oracle1.mpg in the host directory **/home/content/mpeg** and copies both the content files and the created tag file to the MDS volume **video**:

```
% vstag -E mpi -o /mds/video/oracle1.mpg /home/content/mpeg/oracle1.mpg
```

Registering Content Using vsmkosf

These examples use **vsmkosf** to convert an AVI or WAV file into an OSF file and to register the OSF file.

For detailed information about the OSF file format and about how to encode AVI and WAV files that you intend to convert to OSF, refer to Chapter 3, "Encoding Content."

Example 4–19 Converting an AVI File to an OSF File

This example converts and registers the file **video1.avi** located in the local file system and copies the generated OSF and tag file to the MDS volume **video**:

```
c:> vsmkosf ./video1.avi /mds/video/video1
```

> **Note:** You do not need to provide extensions for your output files.

The **vsmkosf** utility generates an OSF file (**video1.osf**) and a tag file (**video1.mpi**) in the MDS, but does not overwrite your source AVI file.

Example 4–20 Converting a WAV File to an OSF File

This example converts and registers the file **audio1.wav** located in the local file system and copies the generated OSF and tag file in the MDS volume **/mds/audio/osf**:

```
% vsmkosf ./audio1.wav /mds/audio/osf/audio1
```

> **Note:** You do not need to provide extensions for your output files.

The **vsmkosf** utility generates an OSF file (**audio1.osf**) and a tag file (**audio1.mpi**) in the MDS, but does not overwrite your source WAV file.

Example 4–21 Converting Multiple AVI Files to OSF Files

This example uses wildcards to convert and register multiple AVI files:

```
% vsmkosf ./*.AVI /mds/video
```

Because this example converts multiple files, the last argument is the output MDS volume, not a file name.

For further information on **vsmkosf**, refer to the *Oracle Video Server Administrator's Guide and Command Reference.*

Registering Content Using vsgentag

These examples use **vsgentag** to register raw key frame or MPEG files with null tag files. Content associated with null tag files do not support rate control operations. (The only operations allowed by a null tag file are normal play and pause). Creating rate-control-enabled tag files for large MPEG files can take a considerable amount of time. For this reason, you may wish to create null tag files with **vsgentag** for MPEG content where the ability to perform rate control is not necessary.

> **Note:** When you register a raw key frame file, you guarantee only that the server will stream the file. **vsgentag** does not provide for playback on the client; the client must be equipped with a codec capable of decoding the incoming stream.

When registering content with **vsgentag**, provide the following information to create a valid tag file:

- bit rate used to encode the file in bits per second.
- compression format of the file. The supported formats are:
  - **key:** specifies raw key frame
  - **mpg1:** specifies MPEG-1
  - **m2t:** specifies MPEG-2 transport (default)
- transport type of the output file. The supported transport types are:
  - **m2t:** specifies MPEG-2 transport (default).
  - **raw:** produces an unframed output file. Specify this when creating null tag files for MPEG-1 content files.

If the above values are not specified, or are incorrect, the file may not play properly or at all.

Example 4–22 Registering an MPEG-1 File with vsgentag

This example creates a null tag file for the MPEG-1 content file **oracle1.mpg**. The values used are:

- bit rate of 1,536,000 bits per second
- MPEG-1 compression format
- raw transport type

```
% vsgentag -f mpg1 -t raw 1536000 /mds/video/oracle1.mpg /mds/video/oracle1.mpi
```

Example 4–23 Registering an Motion-JPEG File with vsgentag

This example creates a null tag file for the Motion-JPEG file **video1.mjpg**. The values used are:

- bit rate of 1,536,000 bits per second
- key compression format
- raw transport type

```
% vsgentag -f key -t raw 1536000 /mds/video/video1.mjpg /mds/video/oracle1.mpi
```

For further information on **vsgentag**, refer to the *Oracle Video Server Administrator's Guide and Command Reference.*

Managing Registered Content

Currently, the command line and VSM interfaces offer different levels of support for managing registered content.

From the command-line interface, you can:

- Display Tag File Contents
- Modify Tag Files
- Update the Content Registry in the Oracle Database

From Video Server Manager, you can:

- Create, Modify, and Remove Logical Content, Clips, and Physical Content

For more information about how to manage registered content from Oracle Video Server Manager, refer to the Video Server Manager online Help.

Displaying Tag File Content

For each content file registered with the OVS system, you can display the contents of a tag file by using the **vstagprint** utility.

Example 4–24 Displaying the Tag File Contents of an MPEG-1 File

This example displays the tag file **oracle1.mpi** in the MDS volume **video**. **oracle1.mpi** is a tag file for an MPEG-1 content file. An explanation of the fields displayed is provided below:

```
% vstagprint /mds.video/oracle1.mpi
Tag file version: 6.1
Current code is version 6.1, back-compatible to version 6.1
magic=aabbccdd (should be aabbccdd)
file describes content for: <Unavailable>
1 member files:
All rate control operations are allowed on this file
# 1. format=MPEG 1     transport=None     flags=1
file="/mds/video/oracle1.mpg" Size = 7167216
creation time of content: Aug  8 17:18:48
bitrate=2048000 bits/second
elapsed length=27996 milliseconds
presentation rate=1000
frames/sec(* 1000)=29970
57 tags in tag file.
width: 352, height: 240, pel aspect ratio (* 10000): 10950
MPEG1 Tagfile version: 1.1
```

```
Current MPEG1 code is version 1.1, back-compatible to 1.1
sequence header is 76 bytes starting at offset 0 of compression data
Tags for member file #1:
1:1, pos= 5315,   PTS- 16262  [002f0005a20b0000000000000000]
2:1, pos= 231383, PTS- 128875 [0b8b0034808d0000000a00000000]
3:1, pos= 471340, PTS- 241487 [0c14003f39930000000c00010000]
4:1, pos= 711650, PTS- 354100 [0c420055891d0000000f00010000]
5:1, pos= 951971, PTS- 466712 [0c7b00570b960000001100020000]
[etc.]
```

The output header tells you:

- the tag format version of the tag file and the versions accepted by the OVS. Different versions of the OVS may expect different tag format versions, so after you upgrade the OVS, this information tells whether you must regenerate your tag file. In this example, both the tag file format and the OVS software are version 6.1, so you need not regenerate.

  Note: The tag format version number is independent of the OVS product version number.

- whether the input file you specified is a valid tag file. If the **magic** value does not match the text following it, the input file is either not a tag file or is corrupted. In this example, the magic value “aabbccdd” matches the text following, so **oracle1.mpi** is a valid tag file.
- the video file described is an MPEG-1 system stream.
- the name and size of the MPEG content file to which the tag file points. In this example, the tag file is based on the content file **/mds/video/oracle1.mpg** which is 7,167,216 bytes in length.
- the bit rate of the associated MPEG content file. In this example, the bit rate of **oracle1.mpg** is 2,048,000 bps.

Each numbered line of output following the header describes a single tag point and has these fields:

integer1:*integer2*	tag number and internal code for the type of the frame tagged
pos	byte position in the MPEG content file where the tagged frame begins
PTS or SCR	Presentation Time Stamp or System Clock Reference—describes when the tagged frame appears as the video is playing
[ ]	contains the information necessary to reposition the video to this tag point as it is playing

Example 4–25 Displaying the Tag File Contents of an OSF File

This example displays the tag file **video1.mpi** in the MDS volume **video**. The **-s** (or short listing) flag shows only the tag file header information; no frame information is listed. **video1.mpi** is a tag file for an OSF content file:

```
% vstagprint -s /mds/video/video1.mpi
Tag file version: 6.1
Current code is version 6.1, back-compatible to version 6.1
magic=aabbccdd (should be aabbccdd)
file describes content for: <Unavailable>
1 member files:
All rate control operations are allowed on this file
# 1. format=Raw Key Frame     transport=None     flags=1
file="/mds/video/video1.osf" Size = 15003788
creation time of content: Aug 11 11:11:47
bitrate=882437 bits/second
elapsed length=136021 milliseconds
presentation rate=1000
frames/sec(* 1000)=10000
272tags in tag file.
width: 240, height: 180, pel aspect ratio (* 10000): 10000
RKF Tagfile version: 1.1
Current RKF code is version 1.1, back-compatible to 1.0
Init data is 88bytes starting at offset 0 of compression data
```

Modifying Tag Files

As the OVS content administrator, you might need to change the registry information for a content file. For example, you might want to move content files to another video server volume, or change the supported rate control operations for a particular file. Rather than re-register your content file, which can be a time-consuming task, you can use **vstagpatch** is to edit the information stored in the tag file.

You can use **vstagpatch** to change the following tag file properties:

- the name of its associated MPEG or OSF content file
- descriptive comment
- bit rate (not recommended)
- restricted rate control operations

Example 4–26 Updating File Names with vstagpatch

In this example, **oracle1.mpi** is the tag file for **oracle1.mpg**. This example uses the **mdscopy** utility to copy these files from one MDS volume to another and then uses **vstagpatch** to associate the new tag file with the new content file:

```
% mdscopy /mds/video/oracle1.mpg /mds/new_clips/oracle1.mpg
% mdscopy /mds/video/oracle1.mpi /mds/new_clips/oracle1.mpi
% vstagpatch -n "/mds/new_clips/oracle1.mpg" /mds/new_clips/oracle1.mpi
```

Example 4–27 Updating the Rate Control Prohibitions with vstagpatch

This example modifies the tag file **oracle1.mpi** in the volume **video**, changing the rate control prohibitions to prevent the viewer from pausing the video during playback:

```
% vstagpatch -D p /mds/video/oracle1.mpi
```

Updating the Content Registry in the Oracle Database

In this release of the Oracle Video Server, content files can be registered with the Oracle MDS as well as the Oracle database.

> **Note:** The OVS content model, described in Chapter 1, assumes that each registered content file is registered only once with the Oracle MDS and database. **Although OVS does not prevent you from creating multiple registries, Oracle strongly recommends that you do not register a content file more than once, as it makes content management extremely complex and difficult.**

Since there are several loading and registering methods available in the OVS system, it is possible for the information in the Oracle MDS and database to get out of sync.

The Oracle MDS and database can get out of sync in the following scenarios:

- content files were registered only with the Oracle MDS, because no database was available to the OVS system
- content files were deleted from the Oracle MDS using the OVS command-line utilities, but the database objects and tag files are still present.

When deleting logical content, clips, or physical content from the database with Video Server Manager, VSM does not permit you to perform any action that will result in an invalid entry in the database. **For this reason, Oracle recommends using VSM for all content management operations, as it ensures that the MDS and the database remain consistent.**

If you choose not to maintain the integrity of your content with VSM, the OVS provides the following command-line utilities:

- **vscontdel**: use this utility to delete database entries for content files that are no longer stored in the MDS.
- **vscontreg**: use this utility to register tagged content files with the Oracle database.

Deleting Obsolete Content Information from the Oracle Database

If you delete content files and/or tag files from the MDS, you must also delete the logical content titles and clips in the database associated with those files. Otherwise, your database will become populated with rows of useless data.

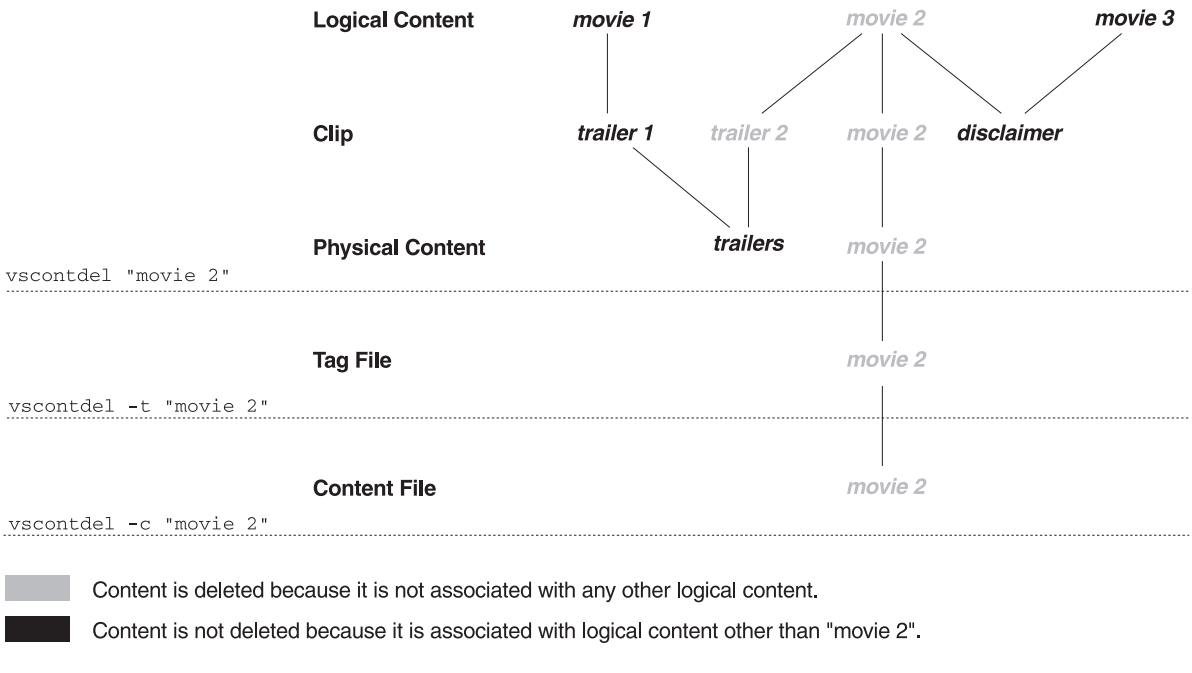
When you use **vscontdel** to delete a logical content title, **vscontdel** performs the following operations:

- It deletes the logical content title.
- It attempts to delete all the clips associated with the logical content title. If a clip is used in at least one other logical content title, the clip will not be deleted. **vscontdel** will print a list of the clips that could not be deleted.
- It will then attempt to delete all the physical content objects associated with the clips that were successfully deleted. If a physical content object is used by at least one other clip, the physical content object will not be deleted. **vscontdel** will print a list of the physical content objects that could not be deleted.
- If a physical content object is successfully deleted **and** if the **-t** or **-c** flag is specified in the command, **vscondel** will then attempt to delete:
  - the tag file associated with the physical content object (**-t**)
  - the tag file and content file associated with the physical content object (**-c**)

WARNING: Use the -t and -c flags with caution. vscontdel cannot determine if a tag file or content file is associated with other physical content objects. Therefore, it is possible to delete tag files and content files that other logical content, clip, and physical content objects still require.

Figure 4–1 illustrates the process **vscontdel** follows when deleting logical content.

Figure 4–1 How vscontdel Attempts to Delete a Logical Content Title and Its Associated Content



The following examples show how to delete obsolete content objects from the database storing your logical content information.

Example 4–28 Deleting a Logical Content Title Using vscontdel

This example deletes the logical content title “Best Cartoons” from the OVS database storing logical content information.

```
% vscontdel "Best Cartoons"
```

Example 4–29 Deleting a Logical Content Title and Tag File Using vscontdel

This example uses **-t** to delete the logical content title “Best Cartoons” from the OVS database storing logical content information as well as the tag file stored in the MDS.

```
% vscontdel -t "Best Cartoons"
```

Example 4–30 Deleting a Logical Content Title, Tag File, and Content File Using vscontdel

This example uses **-c** to delete the logical content title “Best Cartoons” from the OVS database storing logical content information as well as the tag file and content file stored in the MDS.

```
% vscontdel -c "Best Cartoons"
```

Registering Tagged Content with the Oracle Database

You can use the **vscontreg** utility to update logical content entries when moving content from one OVS system to another. For example, if you migrate your OVS system to a more powerful server with an expanded MDS file system, you can retain your existing logical content database by re-registering your tag files with **vscontreg**.

With **vscontreg**, you can:

- add a logical content database to an existing OVS system
- move your OVS installation from one operating system type to another (i.e. Sun Solaris 2.5.1 to HP-UX 10.01)

Example 4–31 Registering a File with the Database

This example reregisters the file **oracle1.mpi** stored in the MDS volume **video**:

```
% vscontreg /mds/video/oracle1.mpi
```

Example 4–32 Registering Multiple Files with the Database

This example reregisters all tag files ending with the extension **.mpi** in the MDS volume **video**:

```
% vscontreg /mds/video/*.mpi
```

Example 4–33 Rewriting a File’s Database Entry

This example uses **-d** to reregister the file **oracle1.mpi** while deleting any existing database objects that use the same name that the tag file uses:

```
% vscontreg -d /mds/video/oracle1.mpi
```

Managing Logical Content with Video Server Manager

Logical content is a collection of video clips that are played in a pre-defined sequence. For example, the logical content title “Binky Bopper Cartoon Show” might include these clips:

- Binky title
- Cartoon 1
- Cartoon 2
- Commercial 1
- Commercial 2
- Cartoon 3
- Promo
- End credits

Each clip represents a start time and stop time within a specific content file. The clip may include the entire content file, or only part of it. In the Binky Bopper example, the three cartoon clips could have been created from a single content file titled **BinkysBest.mpg**.The logical content (or play list) is the sequence in which the collection of clips is played.

From Video Server Manager, you can:

- Create logical content
- Modify logical content
- Remove logical content

Creating Logical Content

To create a logical content title from Video Server Manager:

1. Click the **Create Logical Content** button in the toolbar, or select **Tools | Create Logical Content** from the menu bar. The Create Logical Content dialog box appears in the Detail pane.
2. Fill in the appropriate fields.
3. Click **OK**.

> **Note:** When creating a logical content title, you must choose clips that were created from content files encoded at the same bit rate and with the same codec.

For more detailed information about this procedure, refer to the "Creating Logical Content" topic in the VSM online Help.

Modifying Logical Content

To modify logical content:

1. Expand the video server folder in the Navigator tree.
2. Expand the Content and Logical Content folders.
3. Select the logical content title that you want to modify.
4. Click **Modify** in the toolbar. The Modify Logical Content dialog box appears in the Detail pane.
5. Edit the appropriate fields.
6. Click **OK**.

> **Note:** When modifying a logical content title, you must choose clips that were created from content files encoded at the same bit rate and with the same codec.

For more detailed information about this procedure, refer to the "Modifying Logical Content" topic in the VSM online Help.

Removing Logical Content

To remove a logical content title:

1. Expand the video server folder in the Navigator tree.
2. Expand the Content and Logical Content folders.
3. Select the logical content title that you want to remove.
4. Click **Delete** in the toolbar. The **Delete Logical Content** dialog box appears.
5. Fill in the appropriate fields.
6. Click **OK**.

For more detailed information about this procedure, refer to the "Removing Logical Content" topic in the VSM online Help.

Managing Clips with Video Server Manager

A clip is a logical excerpt from a content file. Each clip corresponds to only one content file. Clips map to a specific start and stop position (in seconds) within a content file.

Example: Your company videotapes and encodes all of its product training seminars. Each seminar exists in its own physical file. To make it easier for employees and customers to view the training seminars, you create a logical content title (a play list) called "Product Training", which is composed of several clips. Clips in this logical content title might include:

- company profile
- message from president
- product A training seminar
- product B training seminar
- customer success stories

Each clip references a specific time segment from a single content file. Different clips within the logical content title can refer to the same or different content file.

From Video Server Manager, you can:

- Create clips
- Modify clips
- Remove clips

Creating Clips

To create a clip:

1. Click the **Create Clips** button in the toolbar, or select **Tools | Create Clips** from the menu bar. The Create Clip dialog box appears.
2. Fill in the appropriate fields.
3. Click **OK**.

For more detailed information about this procedure, refer to the "Creating Clips" topic in the VSM online Help.

Modifying Clips

To modify the settings for a clip:

1. Expand the video server folder in the Navigator tree.
2. Expand the Content and Clips folders.
3. Select a clip title.
4. Click **Modify** in the toolbar. The Modify Clip dialog box appears.
5. Edit the appropriate fields.
6. Click **OK**.

For more detailed information about this procedure, refer to the "Modifying Clips" topic in the VSM online Help.

Removing Clips

To remove a clip:

1. Expand the video server folder in the Navigator tree.
2. Expand the Content and Clips folders.
3. Select the clip that you want to remove.
4. Click **Delete** in the toolbar. The **Delete Clips** dialog box appears.
5. Fill in the appropriate fields.
6. Click **OK**.

For more detailed information about this procedure, refer to the "Removing Clips" topic in the VSM online Help.

Managing Physical Content with Video Server Manager

Physical content is an abstraction of the tag file stored in the Oracle Media Data Store (MDS). Each physical content entry contains the metadata for a piece of content, such as encoding rate, encoding format, the MDS tag file associated with the content file, and the size of the content file. This metadata is the same information that is stored in the MDS tag file. One physical content entry is created each time you register a content file into the MDS and database.

From Video Server Manager, you can:

- View physical content descriptions
- Remove physical content descriptions

Viewing Physical Content Descriptions

To view the description for a specific physical content object:

1. Expand the video server folder in the Navigator tree.
2. Expand the Content and Physical Content folders.
3. Select the physical content title that you want. The General property sheet appears in the detail pane.
4. Click **OK**.

For more detailed information about this procedure, refer to the “Viewing Physical Content Descriptions” topic in the VSM online Help.

Removing Physical Content Descriptions

To remove physical content:

1. Expand the video server folder in the Navigator tree.
2. Expand the Content and Physical Content folders.
3. Select the physical content title that you want to remove.
4. Click **Delete** in the toolbar. The **Delete Physical Content** dialog box appears.
5. Fill in the appropriate fields.
6. Click **OK**.

For more detailed information about this procedure, refer to the “Removing Physical Content” topic in the VSM online Help.

Managing Programs with Video Server Manager

A program is an OVS object used in scheduling broadcast applications like pay-per-view (PPV), near video-on-demand, and regular TV viewing.

> **Note:** In order to use the VSM program scheduler, you must make sure that the Scheduling Services (**vsbcastsrv**, **vschdsrv**, and **vsnvodsrv**) and the Content Service (**vscontsrv**) are up and running on your video server. You can check the status of these services from the video server General property sheet. Refer to the VSM online help or Chapter 9 of the *Oracle Video Server Administrator's Guide and Command Reference* for information on how to start and stop these services.

Using the program scheduler from VSM, you can schedule any logical content title to be played on any available channel at any time. Specifically, you can:

- Create a program schedule
- Edit a program schedule
- Remove a program schedule

A schedule is an OVS object that associates logical content with a delivery channel and a delivery time. For example, you might create a schedule to play *The Binky Bopper Cartoon Show* on channel 3 at 8 AM.

A channel is an OVS object which simulates a traditional broadcast channel. Channels consist of a name, channel number, and physical network address. OVS uses the network address to determine where to stream the video. The name and channel number are used as an easy labeling convention.

Creating a Program Schedule with VSM

To create a program schedule:

1. Click the **Program** button in the toolbar. The Program dialog box appears.
2. Click the **Schedule** tab. The Schedule property sheet appears.
3. Click the **View Schedule** button.
4. Click **Add**. The Add Schedule dialog box appears.
5. Fill in the appropriate fields.

6. Click **OK**.

For more detailed information about this procedure, refer to the "Creating Program Schedules" topic in the VSM online Help.

Modifying a Program Schedule with VSM

To modify an existing program schedule:

1. Click the **Program** button in the toolbar. The Program dialog box appears.
2. Click the **Schedule** tab. The Schedule property sheet appears.
3. Enter the date and time for the schedule you want to edit and click the **View Schedule** button.
4. Select the time slot that contains the schedule in the program guide that you want to edit.

> **Note:** It is possible to have more than one logical content title assigned to a time slot. The VSM Schedule property sheet lists only the logical content title that was most recently programmed.

5. Click **Time Slot Detail**. The Time Slot Detail dialog box appears.
6. Click **Edit**.
7. Edit the appropriate fields.
8. Click **OK**.

> **Note:** Each scheduled instance of a logical content title in the program guide is its own entry in the database. Therefore, you cannot edit all scheduled instances in one step. You must edit each scheduled instance individually.

For more detailed information about this procedure, refer to the "Modifying Program Schedules" topic in the VSM online Help.

Removing a Program Schedule with VSM

To remove an existing program schedule from the program guide:

1. Click the **Program** button in the toolbar. The Program Schedule dialog box appears.
2. Click the **Schedule** tab. The Schedule property sheet appears.
3. Enter the date and time for the schedule you want to remove and click the **View Schedule** button.
4. Select the time slot that contains the schedule in the program guide that you want to remove.

> **Note:** It is possible to have more than one logical content title assigned to each time slot. The VSM Schedule property sheet lists only the logical content title that will air first.

5. Click **Time Slot Detail**. The Time Slot Detail dialog box appears.
6. Click **Remove**. The Remove dialog box appears.
7. Select the schedule(s) to delete.
8. Click **OK**.

> **Note:** Each scheduled instance of a logical content title in the program guide is its own entry in the database. Therefore, you cannot remove all scheduled instances in one step. You must remove each scheduled instance individually.

For more detailed information about this procedure, refer to the “Removing Program Schedules” topic in the VSM online Help.

A

Oracle Media Data Store Naming Conventions and FTP Usage Notes

The Oracle Media Data Store (MDS) is a real-time file system for storing and delivering uninterrupted video in real time. Before loading and managing your content with the Oracle Video Server system as described in Chapter 4, "Loading and Managing Digital Content", you need to understand how the Oracle MDS treats volume and file names.

This appendix provides the following information:

- Naming MDS Volumes and Files
- FTP Usage Notes

Naming MDS Volumes and Files

Oracle MDS volume and file names are case-retentive but not case-sensitive. For example:

- Once you create an MDS volume named **Video**, the OVS utilities always return the name as "**Video**", rather than "**VIDEO**" or "**video**". However, you cannot create another volume named **VIDEO** or **video**.
- If you copy the file **oracle.mpg** into the **Video** volume, **mdsdirsrv** and its clients always return its name as "**oracle.mpg**". However, copying **Oracle.mpg** into the same volume overwrites **oracle.mpg**.
- The only legal characters for new file names are:
  - letters A–Z, both upper and lowercase
  - numbers 0–9
  - underscore (_), hyphen (–), and period (.)

If you have an MDS volume created prior to Release 2.1, which uses earlier file naming conventions, you will still be able to read those files with their previous names.

Referencing Files on the MDS Filesystem

To reference a file contained in the MDS filesystem, specify **/mds** followed by the volume name and the file name. For example:

`/mds/`*`volume_name/filename`*

where:

mds	specifies that the file is contained in the MDS filesystem
volume_name	specifies the MDS volume in which to find the file
filename	the name of the file being referenced

Example A–1 Referencing a File in the MDS Filesystem

This example references the file **oracle1.mpg** located in the MDS filesystem in the volume **video**:

```
/mds/video/oracle1.mpg
```

Referencing Files on a Host Filesystem

To reference a file contained on a host filesystem, specify the path name followed by the file name. For example:

/pathname/filename

where:

pathname	locates a file, directory, or any other object in the host filesystem
filename	the name of the file being referenced

Example A–2 Referencing a File in the Working Directory

This example locates the file **oracle1.mpg** in the directory path **/home/video**:

```
/home/video/oracle1.mpg
```

Example A–3 Referencing a File Relative to the Working Directory

To locate a file and its path relative to the current working directory, prepend a period (.) to the path name. For example:

```
./video/oracle1.mpg
```

Example A–4 Referencing a File Relative to the Parent Directory

To locate a file and its path relative to the directory above the current working directory, prepend two periods (..) to the pathname. For example:

```
../home/video/oracle1.mpg
```

FTP Usage Notes

Some of the procedures presented in Chapter 4, "Loading and Managing Digital Content" require you to use FTP commands. This section provides important information that you should be familiar with before you FTP files into or out of the Oracle MDS.

Referencing Volumes with MDS FTP

When you first connect to the MDS, the current working directory is **/mds**, the root of the MDS file system. It is not possible to change to any directory above this. All MDS volumes available on the system are listed in this directory. You can change the current volume to **/mds/***volume_name* for any existing volume, where the files in that volume may be seen. You can also use the standard UNIX "..", ".", "../*name*", and "./*name*" conventions for navigation.

Note: The FTP **mkdir** and **rmdir** directory commands are not implemented. Although they resemble UNIX directories, MDS volumes are not directories and cannot be manipulated in this manner. However, you can create, rename, and delete MDS files in a particular volume.

Supported File Types

Most FTP clients start in ASCII mode by default. However, the MDS FTP server supports only binary file transfers. Before transferring file, you must use the **binary** command to put the client into binary mode.

Sending the allo Command

In order to create a file in the MDS, you must provide the MDS file system with the maximum size of the file to be transferred. To specify this value manually, you must issue the **allo** command to the MDS FTP server.

The following examples show how to set **allo** for FTP clients on various platforms:

UNIX

```
quote allo number
```

Microsoft Windows 3.1 and Windows 95

```
literal allo number
```

Apple Macintosh

Macintosh FTP clients have a pull-down menu with the entry **send ftp command**. This brings up a command dialog box where you enter:

`allo` *number*

where:

number is the size in bytes of the largest file you intend to transfer.

Index

A

administrative tasks
  archiving content
    about, 4-11
    with FTP, 4-12, 4-14
    with mdstar, 4-12
  copying content
    with mdscopy, 4-9
    with Video Server Manager, 4-9
  creating program schedules, 4-36
  displaying tag file contents, 4-23 to 4-25
  loading content
    about, 4-4
    with FTP, 4-7, 4-8
    with mdscopy, 4-6, 4-7
    with mdstar, 4-5, 4-6
    with Video Server Manager, 4-5
  managing clips, 4-33, 4-34
  managing logical content, 4-31, 4-32
  managing physical content, 4-35
  modifying program schedules, 4-37
  modifying tag files, 4-26
  registering content, 4-17 to 4-22
    about, 4-17
    with Video Server Manager, 4-18
    with vsgentag, 4-21, 4-22
    with vsmkosf, 4-20
    with vstag, 4-18, 4-19
  registering tagged content, 4-30
  removing program schedules, 4-38
  updating content registry in database, 4-27 to 4-30
  verifying integrity of MPEG-2 files, 4-16
analog video
  cameras, 2-4
  equipment, 2-4
archiving content, 4-11 to 4-14
  about, 4-11
  with FTP, 4-12, 4-14
  with mdstar, 4-12
audience, xii
AVI
  defined, 3-17
  encoding tips, 3-22, 3-23
AVI files
  converting and registering, 4-20
  encoding, 3-17 to 3-23
  registering, 4-20

B

broadband digital video applications, 2-2

C

clips
  creating, 4-33
  defined, 1-2
  managing with Video Server Manager, 4-33, 4-34
  modifying, 4-34
  removing, 4-34
codecs
  defined, 3-4
  MPEG, 3-5
  supported by OVS, 3-5
  supported by the NC, 3-19

commands
mdscopy, 4-6, 4-7, 4-9
mdstar, 4-5, 4-6, 4-12
vscontdel, 4-27, 4-30
vscontreg, 4-27, 4-30
vsgentag, 4-17, 4-21, 4-22
vsmkosf, 4-17, 4-20
vsmpegchk, 4-16
vstag, 4-17, 4-18, 4-19
vstagpatch, 4-26
vstagprint, 4-23, 4-24, 4-25
comments, xiv
contact information, xiv
container
Audio Visual Interleave (AVI), 3-6
defined, 3-4
MPEG-1 system, 3-6
MPEG-2 transport, 3-6
Oracle Streaming Format (OSF), 3-6
raw key frame (RKF), 3-6
supported by OVS, 3-6
Waveform (WAV), 3-6
content
archiving, 4-11 to 4-14
archiving with mdstar, 4-12
copying, 4-9 to 4-11
copying with mdscopy, 4-9
copying with Video Server Manager, 4-9
deleting, 4-15
encoding samples, 3-13, 3-21, 3-23
file naming, A-2
managing, 4-23 to 4-38
referencing files in MDS, A-2
registering, 4-17 to 4-22
content file
defined, 1-2
content formats
supported by OVS, 3-6
content model, 1-2 to 1-3
conventions, xiii
copying content
about, 4-9
with mdscopy, 4-9
with Video Server Manager, 4-9

D

database registry
adding new information, 4-30
deleting obsolete content information, 4-28, 4-30
deleting
clips, 4-34
content files, 4-15
logical content, 4-33
obsolete database objects, 4-28
physical content, 4-35
program schedules, 4-38
digital video
cameras, 2-4
equipment, 2-4
identifying the project, 2-2
sample applications, 2-2

E

editing source video, 2-6
encoding content, 3-2 to 3-24
encoding formats
supported by Oracle Video Client, 3-7
encoding guidelines
AVI, 3-17 to 3-23
MPEG, 3-8 to 3-16
WAV, 3-23 to 3-24
encoding methods, 3-2
encoding samples
AVI, 3-21
MPEG, 3-13
encoding stations
purchasing, 3-3 to 3-4
encoding tips
AVI, 3-22 to 3-23
end, ix
enterprise digital video applications, 2-3
equipment
analog video, 2-4
digital video, 2-4
planning and purchasing, 2-4

F

filenames, A-2
files
  AVI, converting and registering, 4-20
  extracting tar files, 4-5, 4-6
  loading with mdscopy, 4-6, 4-7
  naming, A-2
  naming conventions, A-2
  referencing files in MDS, A-2
  referencing files on a host filesystem, A-3
  WAV, converting and registering, 4-20
framework
  defined, 3-4
FTP
  archiving, 4-12, 4-14
  loading content into MDS, 4-7, 4-8
  using to access MDS, A-4 to A-5

I

I-frame regularity, 3-14, 3-16

K

key frame
  frequency in AVI files, 3-22

L

loading content
  about, 4-4
  with FTP, 4-7, 4-8
  with mdscopy, 4-6, 4-7
  with mdstar, 4-5, 4-6
  with Video Server Manager, 4-5
logical content
  creating, 4-32
  defined, 1-2
  managing with Video Server Manager, 4-31, 4-32
  modifying, 4-32
  removing, 4-32
low bit rate digital video applications, 2-3

M

maipulating sequence headers, 3-14, 3-16
management interfaces
  command line, 1-4, 4-2
  Video Server Manager, 1-4, 4-2
managing registered content, 4-23 to 4-38
manipulating PAT and PMT, 3-15
manual
  conventions, xiii
  intended audience, xii
  related documents, xii
  structure, xii
MDS files
  extracting tar files, 4-5, 4-6
  loading with mdscopy, 4-6, 4-7
MDS volumes
  loading files, 4-4, 4-5, 4-6, 4-7
  naming, A-2
mdscopy
  copying content with, 4-9
  loading content with, 4-6, 4-7
mdstar
  archiving content with, 4-12
  loading content with, 4-5
MPEG
  codecs, 3-5
  defined, 3-8
MPEG files
  encoding, 3-8, 3-8 to 3-16
  registering, 4-18, 4-19
MPEG-1
  encoding tips, 3-14
MPEG-2
  encoding tips, 3-15
  verifying file integrity, 4-16

N

NC clients
  supported codecs, 3-19
near video-on-demand (NVOD) schedules
  creating with Video Server Manager, 4-36
  modifying with Video Server Manager, 4-37
  removing with Video Server Manager, 4-38

O

Oracle Streaming Format (OSF) files
  creating, 4-20
  registering, 4-18, 4-19
Oracle Video Client
  supported encoding formats, 3-7
Oracle Video Server
  content model, 1-2 to 1-3
  management interfaces, 1-4
  running without a database, 1-3

P

PAT and PMT manipulation, 3-15
physical content
  defined, 1-2
  managing with Video Server Manager, 4-35
  removing, 4-35
  viewing descriptions, 4-35
planning and purchasing equipment, 2-4
program schedules
  creating with Video Server Manager, 4-36
  modifying with Video Server Manager, 4-37
  removing with Video Server Manage, 4-38
purchasing an encoding station, 3-3 to 3-4

R

rate control
  defined, 4-17
raw key frame (RKF) container, 3-6
recording source video, 2-5
registering
  defined, 1-2
registering content, 4-17 to 4-22
  rate control options, 4-17
  with Video Server Manager, 4-18
  with vsgentag, 4-21
  with vsmkosf, 4-20
  with vstag, 4-18, 4-19
related documents, xii

S

sample digital video applications
  broadband, 2-2
  enterprise, 2-3
  low bit rate, 2-3
sample encoding values
  AVI, 3-21
  MPEG, 3-13
  WAV, 3-23
send us your comments, ix
sequence headers
  manipulating, 3-14, 3-16
software
  encoding, 3-3
  video editing, 2-6
source video
  editing, 2-6
  recording, 2-5
stream sequence. See logical content
structure, xii

T

tag files
  defined, 1-2, 4-17
  displaying contents, 4-23 to 4-25
  modifying with vstagpatch, 4-26
tag files. See also registering content
tar archives
  extracting with mdstar, 4-5, 4-6
tips
  editing source video, 2-6
  encoding AVI files, 3-22
  encoding MPEG-1 files, 3-14
  encoding MPEG-2 files, 3-15
  for recording source video, 2-5

V

video
  loading, 4-4, 4-5
  registering content, 4-17 to 4-22
video cameras
  analog, 2-4
  digital, 2-4

recommended accessories, 2-5
recommended features, 2-4
video editing software, 2-6
Video Server Manager
creating program schedules, 4-36
managing clips with, 4-33, 4-34
managing logical content with, 4-31, 4-32
managing physical content with, 4-35
modifying program schedules, 4-37
registering content with, 4-18
removing program schedules, 4-38
vscontdel
updating database registry, 4-27, 4-30
vscontreg
updating database registry, 4-27, 4-30
vsgentag
registering content with, 4-17, 4-21, 4-22
vsmkosf
registering content with, 4-17, 4-20
vsmpegchk, 4-16
vstag
registering content with, 4-17, 4-18, 4-19
vstagpatch
modifying tag files, 4-26
vstagprint
displaying tag file contents with, 4-23, 4-24, 4-25

W

WAV files
converting and registering, 4-20
encoding, 3-23 to 3-24
registering, 4-20