

What's New in Oracle9i *interMedia* Release 2

An Oracle White Paper
June 2002

What's New in Oracle9i *interMedia*

EXECUTIVE OVERVIEW	3
IMPROVED JAVA SUPPORT	3
Integration with Oracle JDeveloper	3
JAI (Java Advanced Imaging).....	4
JMF (Java Media Framework) Support.....	4
JDBC and Java Classes for Servlets and JSPs	5
<i>INTERMEDIA</i> UTILITY ENHANCEMENTS	5
Callable Annotation Services.....	5
<i>interMedia</i> Clipboard	6
IMAGE CONTENT BASED RETRIEVAL	6
RELATIONAL INTERFACES TO <i>INTERMEDIA</i> SERVICES.....	7
NEW OBJECT TYPE - ORDDOC	7
WEBTOP INTEGRATION.....	8
Integration with Oracle Portal	8
Third Party Web Authoring Tool Integration	8
Integration with Oracle's internet File System (iFS)	8
MEDIA CONTENT REPLICATION.....	9
SQL DIRECT PATH LOADING OF MEDIA OBJECTS	9
EXPORT MEDIA CONTENT TO OPERATING SYSTEM FILES...	9

What's New in Oracle9i *interMedia*

EXECUTIVE OVERVIEW

The goal for the Oracle9i release of *interMedia* was to make it easier for all types of developers to use *interMedia*. Major features that benefit developers are Java support for all types of applications, Oracle Portal integration for developing portals, integration with third party tools for web site development, and various enhancements to *interMedia*'s core services.

IMPROVED JAVA SUPPORT

Integration with Oracle JDeveloper

The Oracle9i JDeveloper Business Components for Java (BC4J) framework supports the Oracle9i *interMedia* object types and integrates seamlessly with *interMedia* at multiple levels.

In the BC4J middle tier, *interMedia* provides a set of BC4J domain classes that extend from the *interMedia* Java Client Classes. These domain classes have inherited all the multimedia retrieval, upload, and manipulation methods from their parents. They also have implemented the required BC4J interfaces so they can be used by the BC4J View Objects and Entity Objects to provide built-in, integrated multimedia capabilities inside the BC4J framework. The *interMedia* BC4J domain classes support both BC4J local mode and remote mode deployment. Users are able to write BC4J tier-independent multimedia Java applications using the *interMedia* domain classes.

In the BC4J client tier, *interMedia* provides support to both JSP application development and Java client development. A set of *interMedia* JSP tags can be used in JSP pages to render image, audio, and video content retrieved from the database and to upload multimedia content into the database. Used with other JSP tags in the BC4J Tag Library, users can write powerful and flexible JSP applications using multimedia from the database.

For the development of Java client applications and applets, *interMedia* provides a multimedia JClient control called `ORDMediaControl`, a Swing-based control that binds to BC4J data sources. `ORDMediaControl` leverages the power of the Java Media Framework (JMF) to render image, audio, and video content retrieved from the database. It also provides buttons for the user to move new media content between the database and the file system and to null out the contents of

the attribute. It also provides a series of VCR buttons so the user can interact with the audio and video content.

Benefit: This approach provides great flexibility in the way in which developers can create new applications. Developers gain all the productivity benefits of BC4J – for example, no programming at all is required to create a JSP application with a standard user interface using the Business Components JSP Application Wizard. The wizard automatically integrates *interMedia* data objects along with standard relational data with no effort required of the developer. Because it is transparent, experienced JDeveloper users have no additional learning curve for creation of media-rich Java applications.

JAI (Java Advanced Imaging)

The Java 2D and Java Advanced Imaging APIs provide a standard method for integrating powerful, image manipulation and functionality into image processing and distribution as well as remote viewing applications. The Java Advanced Imaging package (JAI) provides a broad array of image sample models, color models, and image processing operators and can be extended to include new operators. JAI also includes an easily extensible file codec API that provides a solid foundation of file and compression formats.

Supported methods include:

`BfileInputStream`: a `SeekableStream` that reads from an Oracle BFILE

`BlobInputStream`: a `SeekableStream` that read from an Oracle BLOB

`BlobOutputStream`: an `OutputStream` that writes to an Oracle BLOB

Benefit: Oracle *interMedia* has migrated to Java Advanced Imaging in order to offer a rich, open, platform portable imaging package. Customers benefit from its rich collection of supported file formats and processing operations which have been expanded by the *interMedia* interface and additional supported file codecs, and from the fact that it can run wherever there is a Java Virtual Machine including in the server and in the middle tier.

JMF (Java Media Framework) Support

The Java Media Framework (JMF) is targeted at format support for time based media (audio and video). It is a client side package designed to help thick clients both process and output time based media. *interMedia* provides input and output streams making it easy to connect database audio and video objects to JMF players. The Oracle Custom `DataSource` and `DataSink` is an extension to the current Java Media Framework version 2.0/2.1 developed by Sun Microsystems. This extension allows a JMF application to upload time based media data into an Oracle database, and to access time based media data stored as *interMedia* video/audio objects.

Benefit: The advantage of *interMedia* support for JMF is that it makes it relatively easy for JMF clients to play *interMedia* content with JMF players thus saving development time.

JDBC and Java Classes for Servlets and JSPs

JDBC access to rich content is supported enabling applications to select and operate on a result set that contains sets of *interMedia* columns along with other relational data. Changes were made in JDBC which enable it to return large objects (media) in line with other result set contents and in addition, *interMedia* added a set of Java media classes for servlets and JSPs that greatly reduce the complexity of delivering media from these result sets to browsers.

Benefit: Together, the JDBC changes and the *interMedia* Java classes for servlets and JSPs make Java client side handling of media much simpler and more uniform. The amount of client side *interMedia* Java code is greatly reduced and simplified making it far faster and easier to develop *interMedia* applications.

INTERMEDIA UTILITY ENHANCEMENTS

Callable Annotation Services

interMedia Annotator is now extensible as the Annotator parser Java APIs are exposed for use so that developers can write a custom parser for media source files, or create your own annotation types. Media and document metadata can be extracted, indexed, and mapped to XML documents or database schema through the Annotator's Java APIs. These APIs allow for programmatic invocation of metadata services by any application or scripting language that can use Java APIs including JAVAScript, VBscript, and Apple Script.

Additionally, application developers who need to organize related multimedia files can use the *interMedia* Annotator engine Java APIs to integrate the Annotator functions into an application. Developers can also use the *interMedia* Annotator engine APIs as a tool for bulk loading many multimedia files into the database.

Benefit: Productivity is greatly increased via using these new services. For example, if a large number of movie trailers need to be stored, developers can now write a custom Web-based application to parse the movie trailers, generate annotations for each trailer, and upload the movie trailers to an Oracle database automatically.

Using the extensibility features of Annotator, a developer can easily create a Web-based application can define a new annotation type including media content. The developer can also write a new parser to create the property annotation giving a greater degree of flexibility.

***interMedia* Clipboard**

The *interMedia* Clipboard Release 2 is web technology that allows users to store, organize, search, and retrieve their own personal content out on the web.

Clipboard Release 2 differs from Release 1 in several significant ways: it is a browser based thin client portable across all platforms, accesses the database via the WEBDav protocol, and presents a folder-based interface. It can 'play' media within the browser (via standard browser plugins) and upload contents to the database. It also integrates *interMedia* metadata extraction capabilities into the upload process.

Benefit: A browser-based clipboard for media upload, retrieval and manipulation is now available across a wide variety of operating platforms, lifting requirements for a specific configuration.

IMAGE CONTENT BASED RETRIEVAL

As an image database grows, the difficulty of finding relevant images increases, so *interMedia* now offers content based retrieval for photographic images so that a user can search across a properly indexed image table for images that look 'similar' to a sample, search key image. The information contained in image data is processed and an abstraction of its content in terms of visual attributes is created. Every image inserted into the database is analyzed, and a compact representation of its content is stored as a signature. The signature contains information about visual attributes such as color texture, and shape; images are then matched based on these attributes.

The matching process requires that signatures be generated for the comparison image and each image to be compared with it. Since images are seldom identical, matching is based on a similarity-measuring function for the visual attributes and an associated set of weights. The resulting score is the relative distance between two images being compared. The process operates much like a web text search as it returns in rank order a user-specified number of best matches to the sample image.

Benefit: The primary benefit of using content-based retrieval is reduced time and effort required to obtain image-based information. Image content based retrieval provides a new and powerful search mechanism for finding an image of interest and its associated relational and metadata. So, rather than indexing on the relational data to find an image that may be of interest, the user looks for images that appear visually similar to a sample image. The results are drastically reduced search time as searches may be iterated, and much improved search results.

RELATIONAL INTERFACES TO *INTERMEDIA* SERVICES

In addition to an object interface (attributes and methods), Oracle9i includes a pure relational interface to a set of *interMedia* services so that these services can be used on data stored in BLOBs and BFiles. This Relational Interface gives developers the power of *interMedia* to annotate and manipulate audio, image, and video data stored in BLOBs and BFILES without requiring changes to the existing application schema or instantiation of *interMedia* object types, ORDAudio, ORDVideo, and ORDImage.

Developers can now use static methods of *interMedia* objects with existing and new media stored in BLOBs and BFILES to move media data between the local file system and the database, to parse and extract the properties of the media data, and to store these properties in an XML formatted CLOB and/or in individual relational columns. *interMedia* static methods can also be used to perform image processing operations such as cut, scale, compress, and convert format.

Benefit: This helps many applications that can't deal with objects, as well as many early adopters of BLOBs and BFiles that have reams of media data already stored in the database. With the relational interface, these application do not have to make the investment to convert to objects and they can still make use of *interMedia*'s integral data management facilities.

NEW OBJECT TYPE - ORDDOC

ORDDoc integrates the storage, retrieval, and management of heterogeneous media data in Oracle databases using Oracle9i. *interMedia* ORDDoc can store any heterogeneous media data including audio, image, and video data in the same database column. Instead of having separate columns for audio, image, text, and video objects, one column of ORDDoc objects can be used to represent all types of multimedia.

ORDDoc automatically extracts metadata from data of a variety of popular audio, image, and video data formats. ORDDoc also extracts application attributes and stores them in the comments field of the object in XML. ORDDoc is extensible and can be made to recognize and support other heterogeneous media data formats.

Benefit: The ORDDoc heterogeneous media data type can be used in applications that require storage of different types of media data, such as audio, image, and video, in the same column so a common metadata index can be built on all the different types of media data. Once this index is created, searches can be performed across all the different types of media data. This type of search cannot be done if the different types of media objects are placed in different columns of relational tables.

WEBTOP INTEGRATION

Integration with Oracle Portal

Since *interMedia* objects, including source location information, are stored in Oracle tables, they can be included in the types of data available to Oracle Portal components. Rich content already stored in tables in the database can be easily downloaded or made available in the portal framework by building a form on tables containing *interMedia* objects. Creating a form on a table with *interMedia*-based columns is the same as creating a form on any relational table. Images, audio and video clips can also be uploaded from the desktop directly into any table in the database via an Oracle Portal form.

The Oracle Portal Report component is used to display dynamic data in a columnar report format via a web interface. Portal reports built with either of these wizards can now display information stored as an *interMedia* object type and when needed, buttons are provided to play back audio or video data. Reports containing these objects can also be used as portlets.

The Navigator can be used to browse tables containing *interMedia* objects. The Navigator displays icons with links to the content - it does not embed it onto the page. Users can download or view images from object columns but uploads and updates must be performed via Oracle Portal forms. (Insert, update and delete operations cannot be performed from the Navigator for either reports or forms that contain *interMedia* columns nor will object type attributes display.)

Benefit: Integration with Oracle Portal is transparent, so there is no additional learning curve. This makes it easy for users to develop media rich portlet applications based on *interMedia* content – especially important as Portals and portlets become the means for aggregating information internally and externally.

Third Party Web Authoring Tool Integration

Integration with Macromedia's UltraDev now allows *interMedia* objects to be displayed dynamically at runtime from the Oracle server.

Benefit: UltraDev is the leading content creation environment for the design of web sites that is now connected to Oracle9i for the dynamic display of the rich media objects which keeps the web site current while reducing site maintenance.

Integration with Oracle's internet File System (iFS)

Through its type framework, the internet File System makes it possible to import all media types using *interMedia* parsers to extract metadata and to search for media objects using this metadata.

Benefit: The management of media data types is integral to the internet File System

MEDIA CONTENT REPLICATION

With extended support from Oracle replication, it is now possible to replicate tables that contain media object columns across databases. This is typically done to keep two database in synchronization - the intended purpose is participation in a replication environment.

However, developers can use this approach as an easy way to distribute media objects and data to remote destinations. If the replication objects are not dropped and replication support is not generated, then the table and the data remain at the remote destination site, and any changes at the remote destination site are not replicated. This approach enables developers to distribute a standard database and data set (including media) to a new database environment – a prime example is the synchronization of authoring and production environments for a web site that uses media data.

Benefit: *interMedia* object users benefit as the various *interMedia* objects (ORDimage, ORDaudio, ORDvideo, and ORDdoc) can be replicated across databases. This can be done automatically and globally saving enormous application development costs

SQL DIRECT PATH LOADING OF MEDIA OBJECTS

The SQL*Loader utility provides two approaches to loading data:

- Conventional path loading, which uses the SQL INSERT statement and a bind array buffer to load data into database tables
- Direct path loading, which uses the Direct Path Load API to write data blocks directly to the database on behalf of the SQL*Loader client.

It is now possible for SQL*Loader to move complex datatypes such as media data objects using direct path loading as well as conventional path loading.

Benefit: Direct path loading does not use a SQL interface and thus avoids the overhead of processing the associated SQL statements. Consequently, direct path loading tends to provide much better performance than conventional path loading. Loading of large sets of notoriously large media objects such as images and videos can be done much quicker via direct path load.

EXPORT MEDIA CONTENT TO OPERATING SYSTEM FILES

Now both import and export between media objects and operating system files (external file storage) is possible. EXPORT provides a symmetric method to the

interMedia import method so that media data can now be moved between database storage and local disk files.

Benefit: The value is that applications can now programmatically cause a media object (ORDimage, ORDAudio, ORDvideo, or ORDdoc) to be written from the database to a database local disk file. This is very efficient and makes it easy for an application to make a disk file copy of a media for its own use. As many existing applications continue to have a disk file basis, this feature should prove very useful.



White Paper Title What's new in Oracle9i *interMedia*
June 2002

Author: Shirley Ann Stern
Contributing Authors:

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
www.oracle.com

Oracle Corporation provides the software
that powers the internet.

Oracle is a registered trademark of Oracle Corporation. Various
product and service names referenced herein may be trademarks
of Oracle Corporation. All other product and service names
mentioned may be trademarks of their respective owners.

Copyright © 2001 Oracle Corporation
All rights reserved.