

**Oracle® DIVArchive**  
Export/Import User's Guide  
Release 7.4  
**E73110-02**

September 2016

Oracle DIVArchive Export/Import User's Guide, Release 7.4

E73110-02

Copyright © 2016, Oracle and/or its affiliates. All rights reserved.

Primary Author: Lou Bonaventura

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

---

---

# Contents

<b>Preface</b> .....	vii
Audience .....	vii
Documentation Accessibility .....	vii
Related Documents .....	vii
Conventions .....	vii
<b>1 Introduction</b>	
Overview .....	1-1
New and Enhanced Features and Functionality .....	1-1
<b>2 Operations</b>	
<b>Exporting Tapes</b> .....	2-1
Export Limitations .....	2-2
Export Metadata Parameters .....	2-2
Exported Tape Metadata Files .....	2-3
Export Tapes Procedure .....	2-3
<b>Importing Tapes</b> .....	2-4
Using the Import Command .....	2-5
Import as New Object .....	2-5
Skip Object .....	2-5
Using the Import Date as the Archive Date .....	2-6
Add as an Instance .....	2-6
Error Conditions .....	2-6
Warnings and Limitations .....	2-7
Import Example .....	2-7
Import Tape Procedure .....	2-7
<b>3 Troubleshooting</b>	
Export Failed Error Message .....	3-1
Invalid Parameter Error During Export .....	3-1
Tape Already Exists Error During Import .....	3-1
Unsupported Type Error During Import .....	3-2
Import Process Terminated without Importing .....	3-2

## **4 Frequently Asked Questions**

What is the export XML and FFM file compatibility?.....	4-1
What is the Media Type ID? .....	4-1
What are the unsupported DIVArchive attributes? .....	4-1

## **A Appendix**

Non-Spanning Export XML.....	A-1
Spanning Export XML.....	A-2

## **Glossary**



## List of Tables

2-1	Tape Export Limitation Parameters .....	2-2
2-2	Export Metadata Parameters.....	2-2

---

---

# Preface

This document describes the Export and Import Tape operations using the Oracle DIVArchive 7.4 Control GUI and the operating system's command-line interface.

## Audience

This guide instructs administration and operations personnel in all of the necessary steps to provide full performance of the DIVArchive Export and Import functions.

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

### Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

## Related Documents

For more information, see the *Oracle DIVArchive documentation* set in the *Oracle DIVArchive Core documentation* and *Oracle DIVArchive Additional Features documentation* libraries.

## Conventions

The following text conventions are used in this document:

Convention	Meaning
<b>boldface</b>	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.



---

---

## Introduction

The Oracle DIVArchive Export/Import features enables you to remove one (or more) tapes from one Oracle DIVArchive system and add them to a second DIVArchive system.

### Overview

The *Export* function (on the first DIVArchive site) generates metadata files that describe each tape selected for export, and then ejects the selected tapes from their current tape library.

You use the *Import* function to import the metadata, and then insert the ejected tapes into the second system. The archived objects on the exported tapes are then transferred to the second DIVArchive system.

All export functions and the `Insert Tape` command are executed from the DIVArchive Control GUI. The `Import Tape` function uses the command-line interface. DIVArchive allows more than one set of tapes (whether spanned or not) to be exported to and imported from a single file.

Newly imported objects will have only one instance - the instance residing on the tape(s) that was imported. You also have the option to import an object as an instance of another object already existing in the DIVArchive database. The Import Utility requires your specification of a target tape group for newly imported tape objects. The new objects will belong to the identified tape group and not the tape group of the DIVArchive system from which it was exported.

The Export/Import functionality is compatible with complex objects and has additional fields for the advanced formatting and functionality available in DIVArchive release 7.4.

---

---

**Note:** The exported metadata from a DIVArchive 7.4 export cannot be imported into DIVArchive releases before 7.0. However, exported metadata created from releases of DIVArchive before 7.4 can be imported into the DIVArchive 7.4 system.

---

---

### New and Enhanced Features and Functionality

DIVArchive 7.4 Export/Import includes the following new and enhanced features and functionality:

- The exported XML now contains additional XML attributes named `type` and `elementIds` under the component XML element.

- The `type` attribute uses **D** to represent a directory and **F** to represent a file.
- All components for non-complex objects created before DIVArchive 7.4 will be designated (by default) as files because only files were stored in non-complex objects before this release.
- The `elementIds` attribute presents the fully qualified path of Element ID values for a file or empty folder's fully qualified path.
- All objects created before the DIVArchive 7.4 release will have components that default to a null value for the `elementIds` attribute.

This chapter describes the tape exporting and importing procedures.

## Exporting Tapes

The *Export Tapes* function allows one or more tapes containing DIVArchive objects to be exported for use in another independent DIVArchive system (for example at a remote disaster recovery or partner site).

The metadata of each tape for non-complex objects are maintained in the DIVArchive database. The metadata of each tape is saved to an XML file when the tape(s) are exported and used to transfer the metadata to the other DIVArchive system's database during the import operation.

The metadata for complex objects is maintained in both the DIVArchive database and the metadata database. When an export request is initiated, the Export Utility creates an additional plain text file and assigns a `.ffm` extension to the file.

The export feature checks to see if any of the selected tapes contain objects that span onto other tapes. If so, these tapes are included in a menu so that they can also be exported. These spanned tapes must be selected to export the original list of tapes.

The `Export Tapes` command is not used for transferring tapes between two or more libraries controlled by the same Oracle DIVArchive Manager. To transfer tapes between libraries under the same DIVArchive Manager's control, you use the `Eject` command, move the tape to the desired library, and then execute an `Insert Tape` command.

The default action in the export feature removes the tape metadata from the DIVArchive database after the export. In this case, if an object being exported is the last (or only) instance of the object, it will be removed entirely from the database. However, the object metadata can be left in the original DIVArchive database if desired.

Ejected tapes can also be exported. Ejecting tapes before exporting them is the recommended method when the number of tapes to be exported exceeds the robotic tape library selected Cartridge Access Port (CAP) size.

The media type (Write-Once or not) and whether the media is a cartridge or not is identified in the exported XML file and also imported during an Export/Import operation. The new attributes of the tape element are `isWriteOnce` and `isCartridge` each with a value of either `true` or `false`.

## Export Limitations

Tape export limits are configured in the `manager.conf` configuration file. There are several configurable parameters as described in the following table.

**Table 2–1 Tape Export Limitation Parameters**

Parameter	Definition	Limits
DIVAMANAGER_MAX_EXPORT_TAPES	The maximum number of tapes allowed in an export request. Reloadable in <i>SERVICE</i> mode.	Default value is 10 and the maximum value is 25. Example: DIVAMANAGER_MAX_EXPORT_TAPES=10
DIVAMANAGER_MAX_EXPORT_ELEMENTS	The maximum number of elements allowed in an export request. Reloadable in <i>SERVICE</i> mode.	Default value is 100000 and the maximum value is 100000. Example: DIVAMANAGER_MAX_EXPORT_ELEMENTS=100000

Oracle *highly* recommends:

- Only performing one export operation at a time. You risk data loss if more than one export operation is running simultaneously.
- Not performing large exports during peak periods. System performance will be decreased during large exports.
- Delete and repack actions do not clear WORM drives as these are Write-Once Media. The instances are deleted but the space is not recoverable.

## Export Metadata Parameters

The following table describes the export metadata parameters.

**Table 2–2 Export Metadata Parameters**

Parameter	XML Element and Attribute	Notes
objectId	Attribute of the object element	Not imported - A new Object ID is generated during import.
uuid	Attribute of the object element	Imported if present, otherwise a new UUID will be generated.
format	Attribute of the object element and attribute of the tape element	0 = Legacy 1 = AXF 0.9 2 = AXF 1.0 -1 = Unknown
numFolders	Attribute of the object element	
isHeaderValid	Attribute of the object element	
isComplex	Attribute of the object element	
footerBeginPos	Attribute of the element's element	If exists in the database
footerEndPos	Attribute of the element's element	If exists in the database

**Table 2–2 (Cont.) Export Metadata Parameters**

Parameter	XML Element and Attribute	Notes
compOrderNumBegin	Attribute of the element's element	If exists in the database
compOrderNumEnd	Attribute of the element's element	If exists in the database
fileFolderMetadataInfo	Element	Valid for complex objects
fileFolderMetadataInfo-elem	Element	Valid for complex objects
checksums and checksum	Element	Not valid for complex objects
elementIds	Attribute of the component element	The fully qualified path of Element ID values for a file or an empty folder's fully qualified path.
type	Attribute of the component element	Represents the type of object component: D = Directory F = File  Components of non-complex objects created before the 7.4 release default to F because only files were stored in non-complex objects before release 7.4.

## Exported Tape Metadata Files

The DIVArchive system writes each tape's metadata to a .xml file during the export process. The process generates an additional .ffm file for each exported complex object. If an object is spanned across two (or more) tapes, the XML file will encompass every tape included in the spanned set. The naming format of each tape metadata XML file is Tapeset-<Barcode>.xml (for example Tapeset-000131.xml).

The *Root Path* where the XML files are saved is defined by the DIVAMANAGER\_EXPORT\_ROOT\_DIR parameter in the DIVArchive Manager configuration file. By default the export absolute folder *Root Path* is DIVA\_HOME\Program\Manager\bin\exported\.

From this *Root Path* the .xml and .ffm files (if complex objects exist) from each Export Tapes command are saved in sub-directories based on the date and time the command was run.

The .ffm file contains file and folder information for complex objects. The .ffm files are referenced from within the specified .xml file and are named using the *Object Name* and *Object Category* of the exported object. This file must exist in the same directory as the .xml file when importing. The Import Utility will look for them both in the same location. If the file is missing, the import process will terminate and an error message will be written to the log file.

## Export Tapes Procedure

The Export Tape request is initiated using the **Export Tape** button on the GUI ribbon bar, or the **Tapes** view in the **Home** tab by right-clicking the tape to export and selecting **Export Tape** from the resulting menu. When selecting the tapes for export, it is possible to see more tapes available in the tape window than initially selected. If a

tape has objects that are spanned onto another tape, these tapes are also included. In this case, select all of the spanned tapes from this list for the export to succeed.

Use the following procedure to export tapes:

1. Highlight and then right-click the tapes desired for export.
2. Select **Export Tape** from the context menu to begin the export process.

The Export Tape dialog box will appear showing information about the selected tapes and options for the export process. The available options include:

**Comments**

Enter any comments desired in the text box. They will be stored in the request's properties.

**Delete From DB**

If checked, the barcodes, tapes, and object instances stored on those tapes will be deleted from the DIVArchive database upon completion of the export. This parameter is set to `true` by default.

If tapes or object instances are needed in the system again after they have been exported, you must import them because this option removes them from the system's database.

**Exported Tapes**

This area identifies which tapes were selected from the Control GUI for export, if the tape has the original barcode, and if it can be removed from the export operation. For example, if a tape is part of a tape set (rather than a single tape), the *Can Be Removed* column would indicate `No` for that tape because it is required to complete the export successfully.

**Remove Selected**

Removes the highlighted tapes in the **Exported Tapes** area from the export process.

3. After all options have been set and verified, click **OK** to begin the tape export.

This is a multi-step process. If a set of tapes was selected that includes another spanned tape, the GUI will display re-selection dialogs enabling selection of additional tapes in the set.

When the **OK** button is clicked, the export process begins. This results in a `.xml` (and possibly `.ffm` files) being created in the export folder. The XML and FFM files contain all of the information concerning the objects on the tape(s) being exported.

When the export is complete, a good practice is to compress all of the resulting files into a `.zip` file. You must include all of the files because they are required for the import process to complete successfully.

---

---

**Caution:** When using complex objects, the FFM files must be in the same folder as the XML files for importing. If the FFM files are not found the import process will terminate and an error will be written to the log file.

---

---

## Importing Tapes

Importing tapes to be used in restore operations is a two-step process. First, the metadata that describes the tape objects is imported using the `importtapes`

command-line utility. Once the metadata has been successfully loaded, the physical tapes can be inserted into the tape library using the **Insert** function in the DIVArchive Control GUI.

---

---

**Note:** Performing multiple simultaneous import operations are allowed, but not recommended.

---

---

## Using the Import Command

To use the `importtapes` command you must first ensure that the exported XML metadata file and the `.ffm` files exist on the destination DIVArchive System. The files must exist in uncompressed form in the DIVArchive Manager's `bin` directory (by default). Also, the *Object Tape Group* must already exist on the target system before the import begins. This tape group does not necessarily have to be the same group assigned to the tape in the source system.

The three main ways that a tape object can be treated during the import process are as follows:

- Imported as a new object
- Skipped
- Added as an instance of an object already existing in the DIVArchive database

### Import as New Object

Normally, when a tape object is imported by the utility it is imported as a new DIVArchive object. This can only occur when the *Object Name* and *Object Category* for the tape object does not exist in the target DIVArchive system. In the event of a naming conflict, the default behavior is to terminate the import operation without importing any tapes or objects.

When new objects are imported into the target DIVArchive system, the import function only looks at the XML and FFM files and does not read directly from the tape structure. SPM is also automatically notified and if the object matches any of the SPM filters, then SPM will initiate the required actions for the object.

### Skip Object

---

---

**Caution:** You must be careful when skipping objects because the tape object that is skipped may or may not actually be the same as the object in the database. The tape object that had the naming conflict may in fact contain different content than the existing one in the DIVArchive database (content that should be preserved). If a tape is imported and then repacked, objects that were skipped will not be copied to the new tape and the old tape will be reclaimed. If all objects on a tape are skipped (and the tape is made writable), the tape will be marked for deletion and new objects will overwrite existing objects on the tape. If the last object on a tape is skipped and the new objects are written to the tape, that tape instance will immediately be overwritten.

---

---

A tape object can be skipped if the `-skipIfNameExists` flag is passed to the Import Utility. If there is another object already in the DIVArchive database that has the same *Object Name* and *Object Category* as a tape object being imported, and the `-skipIfNameExists` flag is set, the object is skipped. In this case the object instance on

the tape is not recorded in the DIVArchive database (it is considered deleted by DIVArchive), and processing continues with the next tape object in the import metadata.

### Using the Import Date as the Archive Date

The DIVArchive `TapeImport` command-line utility provides an additional command-line switch named `-useImportDateAsArchiveDate`.

Using this switch during object import causes the date of the imported object to be used as the date of object archival in the system where it is being imported. The original archive date is not replaced in the XML export or on the original DIVArchive system, it is only replaced for the object on the imported system.

---

---

**Note:** This feature supports tapes with spanned objects in the same way as regular tapes.

---

---

### Add as an Instance

An object can be imported as an instance of another object if the `-addAsInstanceIfExists` flag is passed to the Import Utility. If there is another object already in the DIVArchive database that has the same *Object Name* and *Object Category* as a tape object being imported, and the `-addAsInstanceIfExists` flag is passed, an Import as an Instance can be attempted.

First, the checksums for the tape object are compared to the checksums in the database object that matches it. If a match is produced (for each object component), the object is imported as an instance of the matching object. The *Comments*, *Archived Path Root*, *Archive Date*, *UUID*, *Storage Plan*, *Group*, and so on, of the imported object are lost and become that of the object already in the DIVArchive database.

---

---

**Note:** Object instance IDs are neither exported nor imported. A new ID is assigned every time the utility imports as an instance.

---

---

If the *Checksum Type* of the object components in the database does not match the *Checksum Type* in the imported object or if one of the two objects has checksums that are missing, the tape object will not be imported as an instance. This is considered a checksum mismatch and the import processing will halt. However, if both the `-skipIfExists` flag and the `-addAsInstanceIfExists` flag are passed to the Import Utility (and a tape object matches one that already exists in the DIVArchive database), the utility will first try to import the object as an instance by comparing checksums. If this attempt fails the object will be skipped and processing will continue.

---

---

**Note:** SPM is not notified when importing as an instance. If the object matches any of the SPM Filters then SPM will not initiate the required actions for the object.

---

---

### Error Conditions

If the tape media is not recognized by the Manager an error will be generated specifying what occurred.

If the import process fails and Manager detects a database error, the import process will be terminated and any operations performed during the failed import will be rolled back and not saved in the system.

In the case where the checksum comparison failed (or the checksum is not present) for one or several objects, the entire import process will be stopped and the database transaction will be rolled back.

If the `-skipIfNameExists` flag is used, the checksum verification will still execute. However in this case an unverified (mismatched) object will be skipped instead of stopping the entire import process.

All errors are displayed on the screen and written to the log file. When using the `-skipIfNameExists` flag, you must check the screen messages and log file to determine whether all content intended to be imported was processed successfully. This option is not compatible with automated workflows since it may require operator intervention and decision.

### Warnings and Limitations

Complex objects that are compared this way must have been archived in the same exact order to pass the checksum verification.

The Import Utility does not compare *UUID*, *Object ID*, *Archive Dates*, or *Site ID*. The *Comments*, *Archived Path Root*, *Archive Date*, *UUID*, *Storage Plan*, *Group*, and so on, of the imported object are not preserved when being added as an instance.

The utility does not allow the import of a set of tapes that contain an object with more than one instance on the tapes. An import metadata file having an object with more than one instance appearing within an exported tape set is not allowed. The export utility prevents this from happening.

## Import Example

The tape with barcode number *000131* also contains objects that are spanned across the tape with a barcode of *000120*. When tape *000131* is exported, its exported XML File is named `Tapeset-000131.xml`. This XML file also includes the objects from tape *000120*, and both tapes *000131* and *000120* will be ejected from the library. After all objects from both tapes are exported to the XML file, all instances on each tape and references to the tapes themselves are removed from the DIVArchive database.

The XML file is then copied to the `DIVA_HOME\Program\Manager\bin` folder of the target DIVArchive system. The command `importtapes MOVIES Tapeset-000131.xml` results in the metadata for this tape being imported into the group *MOVIES*.

When the tape's metadata has been successfully imported to the database (check the Control GUI *Current Requests* queue), both of the tapes and their objects are considered externalized and can then both be entered into the library with the **Insert Tape** command.

Importing of WORM Media is supported by DIVArchive 7.4 and later. However, if an import of a DIVArchive 7.4 (or later) export containing WORM media into an earlier DIVArchive release, the WORM flag will be ignored (set to `false`) and logged in the Manager log. The device will be seen in the Control GUI as a tape but not usable if finalized or no WORM drive is connected to the system.

## Import Tape Procedure

Importing of tapes is accomplished using a combination of the Windows command-line interface and the DIVArchive Control GUI. Inserting the tape is an optional part of the workflow but is necessary to access the objects on the tape. It is possible to run the `importtape` command-line utility to enter the tape's metadata into the DIVArchive database and still keep the tape externalized. However, to access the

objects on the tape, the tape must be inserted using the DIVArchive insert tape function.

The following procedure is used for importing tapes into DIVArchive:

1. Open a Windows command-line interface.
2. Copy the exported XML and FFM Files into the `DIVA_HOME\Manager\bin` folder.
3. Change to the `DIVA_HOME\Manager\bin` folder.
4. Run the `importtape` command using any of the following necessary command-line options:

**help (-h)**

Displays help information.

**groupname**

The tape group to which imported tapes will belong. The group must already exist in the system.

**mfiledir**

The XML file containing exported tape metadata, or a folder that contains the files.

**-skipIfNameExists**

Skip import of objects with naming conflicts. The default behavior is that if the *Object Name* and *Object Category* already exist, the utility will terminate without importing the tape(s). Using this option in the command-line will override the default.

**-addAsInstanceIfNameExists**

Attempt to add the tape object as an instance of an existing object in the DIVArchive database. The tape object must have the same *Object Name* and *Object Category*, components, and checksums as the object in the database.

**-useImportDateAsArchiveDate**

Changes the imported object's original archive date to the date of import on the destination system. This does not change the original archive date in the exported XML file or in the original system where the object was exported from, only on the system where the object was imported.

5. In the DIVArchive Control GUI, navigate to the **Home** tab, and then click the **Tapes** button to show the list of tapes identified in the system through the **Tapes** panel. Imported tapes can be left externalized, but to restore the objects on a tape it must be inserted into the library.
6. Highlight the desired tape (or tapes) and then navigate to the **Action** tab on the ribbon bar and click **Insert Tape** to open the Insert Tape dialog box.
7. If the object's instance must exist in the database before the tape is inserted, select the check box *Require instances on tape(s)*. Otherwise leave it deselected.
8. Select the appropriate *Robot Manager Name* using the menu.
9. Select the appropriate *CAP ID* using the menu.
10. Use the slide control to select the priority value for the insert operation.
11. Restoration of the objects on the imported tapes is possible after the tapes are inserted.

---

---

## Troubleshooting

This chapter describes basic troubleshooting procedures. Contact Oracle Support for additional assistance when required.

### Export Failed Error Message

```
Robot Manager Error : Error while ejecting tapes: StatusCode[70:INTERNAL_ERROR]
Request step is STEP_WAITING_FOR_OPERATOR()
```

**Resolution:**

Check that the CAP where tapes are being ejected to has not reached its capacity. Even if the CAP is empty, if more tapes than the capacity of the CAP are exported a successful export operation cannot be completed. This is specifically an issue with sets of spanned tapes and the number of tapes in that spanned set is greater than the number of tapes supported by the CAP. In this case, eject the tapes first then perform the export.

### Invalid Parameter Error During Export

```
Invalid parameter : Tape Y00105 must be included into export list
```

**Resolution:**

When selecting the tapes for export, you can possibly see more tapes available in the tape window than initially selected. If a tape has objects that are spanned onto another tapes, these tapes are also included. In this case, select all of the spanned tapes from this list in order for the export to succeed.

### Tape Already Exists Error During Import

```
The following errors were found in tapeset-J00026.xml\Tape J00026 already exists
in DIVA. Consider performing a tape Insert operation...
```

**Resolution:**

A tape with the same barcode as the one being imported already exists in the DIVArchive system. It is likely that the tape metadata for the tape you want to import already exists in the DIVArchive database and you just need to perform an **Insert Tape** operation to use the tape. Verify the tape contains the correct objects by using the DIVArchive Control GUI.

## Unsupported Type Error During Import

The following errors were found in tapeset-[Y00109].xml\Tape Y00109 has unsupported type 19.

### Resolution:

The *type* in the message refers to the `mediaTypeId`. The `mediaTypeId` is an ID that represents the type of tape media being exported. DIVArchive exports a `mediaTypeId` field that corresponds to the *Id* column in the *Tape Properties* table under the **Tapes** tab in the DIVArchive Configuration Utility. You may need to execute a Synchronize DB call to update the `mediaTypeId` and (or) update your hardware to be compatible with a newly imported tape. Ensure that the block size and total size of the `mediaType` in the source DIVArchive system matches the `mediaType` definition in the destination.

## Import Process Terminated without Importing

There are several reasons why the import process may terminate without completing successfully including the following:

- When using complex objects, the FFM files must be in the same folder as the XML files for importing. If the FFM file is not found, the import process will terminate and an error will be written to the log file.
- If the *Object Name* and *Object Category* already exist, and the `-skipIfNameExists` or `-addAsInstanceIfNameExists` options are not passed, the utility will terminate without importing.
- If the Manager detects a database error the import process will be terminated and any operations performed during the failed import will be rolled back and not saved in the system.

---

---

## Frequently Asked Questions

This chapter addresses some frequently asked questions received by customers.

### What is the export XML and FFM file compatibility?

The exported XML and FFM files, when generated, can be imported into the release of DIVArchive that they were exported from, and later releases of DIVArchive.

DIVArchive allows more than one set of tapes (spanned or not) to be exported to and imported from a single file.

Exported metadata from the DIVArchive 7.4 export function cannot be imported into DIVArchive releases earlier than release 7.0. However, exported metadata created from releases of DIVArchive before 7.4 can be imported into the DIVArchive 7.4 system.

### What is the Media Type ID?

The Media Type ID is a proprietary DIVArchive identifier that represents the type of tape media being exported. DIVArchive exports a `mediaTypeId` field, which corresponds to the *Id* column in the **Tape Properties** table under the **Tapes** tab in the DIVArchive Configuration Utility. You may need to execute a `Synchronize DB` call to update the `mediaTypeId`, and (or) update your hardware to be compatible with a newly imported tape. You should ensure that the block size and total size of the `mediaType` in the source DIVArchive system matches the `mediaType` definition in the destination. This becomes especially important if the tape is ever repacked.

### What are the unsupported DIVArchive attributes?

The `markedAsDeleted` is an internal attribute and is not exported or imported through the Export/Import Utility. In addition, the state of checksum verification (verified, partially verified, and so on) is not exported. Linked objects and link information is not exported. Information regarding the request that created each object is not exported - newly imported objects are not associated with a DIVArchive request.

What are the unsupported DIVArchive attributes?

---

This appendix provides sample XML files.

## Non-Spanning Export XML

```
<tapeset class="com.storagetek.diva.messaging.types.ExportedTapeSetMetadata"
exportDate="27 Oct 2010 20:55:30 GMT" divaName="MGR_650" divaVersion="DIVA_6_5_1_
0_0">
  <tapes array-size="1">
    <tape barcode="Y00103" mediaTypeId="13" remainingSizeKB="30803" fillingRatio="3"
fragmentation="0" blockSize="65535" lastWrittenBlock="19" lastArchiveDate="27 Oct
2010 20:55:01 GMT" firstInsertDate="21 Apr 2010 19:02:49 GMT" firstMountDate="27
Oct 2010 20:54:05 GMT" isHeadTape="true" originalGroup="MOV">
      <elements array-size="4">
        <element objectName="TEST" category="SMALL" compNum="1" elemNum="1"
beginPos="2" endPos="5" elemSizeKB="2" stopPos="2371" />
        <element objectName="TEST2" category="SMALL" compNum="1" elemNum="1"
beginPos="7" endPos="10" elemSizeKB="1" stopPos="41" />
        <element objectName="TEST3" category="SMALL" compNum="1" elemNum="1"
beginPos="12" endPos="15" elemSizeKB="1" stopPos="73" />
        <element objectName="TEST3" category="SMALL" compNum="2" elemNum="1"
beginPos="16" endPos="17" elemSizeKB="1" stopPos="72" />
      </elements>
    </tape>
  </tapes>
  <objects array-size="3">
    <object objectName="TEST" category="SMALL" comments=" " sourcename="origin_ftp"
rootOnSource=" " dateArchive="27 Oct 2010 20:54:05 GMT" numComponents="1"
numElements="1">
      <components array-size="1">
        <component name="a1.txt" compNum="1" sizeKB="2" sizeBytes="2372">
          <checksums array-size="1">
            <checksum csValue="40f818c93e17c94fd476951f9f5db788" csSource="AC"
csType="MD5" />
          </checksums>
        </component>
      </components>
    </object>
    <object objectName="TEST2" category="SMALL" comments=" " sourcename="origin_ftp"
rootOnSource=" " dateArchive="27 Oct 2010 20:54:20 GMT" numComponents="1"
numElements="1">
      <components array-size="1">
        <component name="a2.txt" compNum="1" sizeKB="1" sizeBytes="42">
          <checksums array-size="1">
            <checksum csValue="0be6e7d72fdb52266b9c99540b3755ce" csSource="AC" csType="MD5"
/>
          </checksums>
        </component>
      </components>
    </object>
  </objects>
</tapeset>
```

```

    </checksums>
  </component>
</components>
</object>
<object objectName="TEST3" category="SMALL" comments=" " sourcename="origin_ftp"
rootOnSource=" " dateArchive="27 Oct 2010 20:55:01 GMT" numComponents="2"
numElements="1">
  <components array-size="2">
    <component name="a3.txt" compNum="1" sizeKB="1" sizeBytes="74">
      <checksums array-size="1">
        <checksum csValue="b0354657e98cf78074a6409dce2697c8" csSource="AC" csType="MD5"
      />
    </checksums>
  </component>
    <component name="a4.txt" compNum="2" sizeKB="1" sizeBytes="73">
      <checksums array-size="1">
        <checksum csValue="2bfa170db4ada38a27085cb4b339f05e" csSource="AC" csType="MD5"
      />
    </checksums>
  </component>
</components>
</object>
</objects>
</tapeset>

```

## Spanning Export XML

```

<tapeset class="com.storagetek.diva.messaging.types.ExportedTapeSetMetadata"
exportDate="27 Oct 2010 20:44:57 GMT" divaName="MGR_650" divaVersion="DIVA_6_5_1_
0_0">
  <tapes array-size="2">
    <tape barcode="Y00105" mediaTypeId="13" remainingSizeKB="500" fillingRatio="98"
fragmentation="0" blockSize="65535" lastWrittenBlock="500" lastArchiveDate="27 Oct
2010 20:38:59 GMT" firstInsertDate="21 Apr 2010 19:02:49 GMT" firstMountDate="27
Oct 2010 20:38:55 GMT" isHeadTape="true" spannedTo="Y00104" originalGroup="MOV">
      <elements array-size="1">
        <element objectName="BIG2" category="SPAN" compNum="1" elemNum="1"
beginPos="2" endPos="500" elemSizeKB="31679" stopPos="32440080" />
      </elements>
    </tape>
    <tape barcode="Y00104" mediaTypeId="13" remainingSizeKB="14360"
fillingRatio="55" fragmentation="0" blockSize="65535" lastWrittenBlock="280"
lastArchiveDate="27 Oct 2011 20:38:59 GMT" firstInsertDate="21 Apr 2010 19:02:49
GMT" firstMountDate="27 Oct 2010 20:38:59 GMT" isHeadTape="false"
originalGroup="MOV">
      <elements array-size="1">
        <element objectName="BIG2" category="SPAN" compNum="1" elemNum="2"
beginPos="2" endPos="278" elemSizeKB="17443" stopPos="50302194" />
      </elements>
    </tape>
  </tapes>
  <objects array-size="1">
    <object objectName="BIG2" category="SPAN" comments=" " sourcename="origin_ftp"
rootOnSource=" " dateArchive="27 Oct 2010 20:38:59 GMT" numComponents="1"
numElements="1">
      <components array-size="1">
        <component name="Dbig.txt" compNum="1" sizeKB="49122" sizeBytes="32440081">
          <checksums array-size="1">
            <checksum csValue="f53d6dbdaa266a5e7327683f971fcd7d" csSource="AC"

```

```
csType="MD5" />  
  </checksums>  
  </component>  
</components>  
</object>  
</objects>  
</tapeset>
```



---

---

# Glossary

## **AXF or AXF Media Format**

The Archive Exchange Format (AXF) is based on a file and storage media encapsulation approach which abstracts the underlying file system, operating system, and storage technology making the format truly open and non-proprietary. AXF helps ensure long-term accessibility to valued assets, and keeps up with evolving storage technologies.

## **CAP ID**

The designation of a slot in the Tape Library.

## **Complex Object**

An object is defined as a complex object when it contains more than 1,000 components (configurable). Complex object handling may differ from non-complex objects as noted throughout this document.

## **Legacy Format**

Proprietary storage format used in DIVArchive releases 1.0 through 6.5.1.

## **Metadata Database**

The metadata database is the location where the metadata for components of complex objects are stored in the DIVArchive system.

## **Metadata File**

The file listing the *Object Name* and *Object Category* contained on a tape and its location.

## **Non-complex Objects**

DIVArchive objects with 1,000 files or less are considered non-complex objects. The maximum number of files an object can hold is configurable.

## **Robot Manager**

The mechanical tape system used with DIVArchive to insert and eject tapes to and from the tape library.

## **Universally Unique Identifier (UUID)**

A Universally Unique Identifier distinctly identifies each object created in DIVArchive across all Oracle customer sites except for objects created through Copy As requests. An object created using a Copy As request will contain the same UUID as that of the source object.

