223
DBMS_XDB

The `DBMS_XDB` package supports the following features:

> **✎ Note:**
>
> The Oracle XML DB Repository is deprecated with Oracle Database 23ai.
>
> Oracle recommends that you replace any functionality used in XML DB Repository with alternative technologies. As a result of this deprecation, all XML DB Repository interfaces (for example Repository-specific Java classes `oracle.xdb.servlet`, `oracle.xdb.event`, and `oracle.xdb.spi`) are consequently deprecated as well.

The `DBMS_XDB` package supports the following features:

- Resource Management subprograms which complement Resource Views
- The Access Control List (ACL)-based Security Mechanism
- Configuration Session Management
- Creation of the XDB username

This chapter contains the following topics:

- DBMS_XDB Overview
- DBMS_XDB Security Model
- DBMS_XDB Constants
- Summary of DBMS_XDB Subprograms

> **✎ See Also:**
>
> - *Oracle XML DB Developer's Guide*

DBMS_XDB Overview

`DBMS_XDB` provides a range of management functions.

The `DBMS_XDB` package supports the following features:

- The Resource Management functionality provides LINK Procedures, EXISTSRESOURCE Function, LOCKRESOURCE Function, GETLOCKTOKEN Procedure, UNLOCKRESOURCE Function, CREATERESOURCE Functions, RENAMERESOURCE Procedure, DELETERESOURCE Procedure, GETRESOID Function, CREATEOIDPATH Function, and CREATEFOLDER Function subprograms which complement Resource Views.

- The Access Control List (ACL)-based Security Mechanism can be used with in-hierarchy ACLs stored by the database or in-memory ACLs that may be stored outside the database. Some of these methods can be used for both Oracle resources and arbitrary database objects. Use CHECKPRIVILEGES Function, GETACLDOCUMENT Function, and CHANGEPRIVILEGES Function for Oracle Resources. ACLCHECKPRIVILEGES Function provides access to Oracle's ACL-based Security mechanism without storing objects in the Hierarchy.

- The XDB username is created during XDB installation. This user owns a set of default tables and packages. GETXDB_TABLESPACE Function and the `DBMS_XDB_ADMIN.MOVEXDB_TABLESPACE` Procedure enable movement of schemas to a specified tablespace, and support the default `SYSAUX` tablespace introduction

DBMS_XDB Security Model

Owned by `XDB`, the `DBMS_XDB` package must be created by `SYS` or `XDB`. The `EXECUTE` privilege is granted to `PUBLIC`. Subprograms in this package are executed using the privileges of the current user. Subprograms that operate on the XDB Configuration will succeed only if the current user is `SYS` or `XDB`, or the current user has the `XDBADMIN` or `DBA` role.

DBMS_XDB Constants

All constants described in the following table are deprecated in Oracle Release 12*c*.

They are relocated to either the DBMS_XDB_CONFIG package or the DBMS_XDB_REPOS package. The specifics of transference in each case are detailed in the **Relocated** column.

Oracle recommends that you do not use constants in their `DBMS_XDB` context in new applications. Support for deprecated features is for backward compatibility only and may be terminated in future releases.

Table 223-1   DBMS_XDB Constants

Constant	Type	Value	Description	Relocated
`DELETE_RESOURCE`	`NUMBER`	1	Deletes a resource; fails if the resource has children.	`DBMS_XDB_REPOS`
`DELETE_RECURSIVE`	`NUMBER`	2	Deletes a resource and its children, if any.	`DBMS_XDB_REPOS`
`DELETE_FORCE`	`NUMBER`	3	Deletes the resource, even if the object it contains is invalid.	`DBMS_XDB_REPOS`
`DELETE_RECURSIVE_FORCE`	`NUMBER`	4	Deletes a resource and its children, if any, even if the object it contains is invalid.	`DBMS_XDB_REPOS`
`DELETE_RES_METADATA_CASCADE`	`NUMBER`	1	Deletes the corresponding row in the metadata table	`DBMS_XDB_REPOS`
`DELETE_RES_METADATA_NOCASCADE`	`NUMBER`	2	Does not delete the row in the metadata table	`DBMS_XDB_REPOS`
`DEFAULT_LOCK_TIMEOUT CONSTANT`	`PLS_INTEGER`	(60*60)	Default time (in seconds) after which lock will expire	`DBMS_XDB_REPOS`
`LINK_TYPE_HARD`	`NUMBER`	1	Type of link to be created (default)	`DBMS_XDB_REPOS`

Table 223-1   (Cont.) DBMS_XDB Constants

Constant	Type	Value	Description	Relocated
LINK_TYPE_WEAK	NUMBER	2	Type of link to be created	DBMS_XDB_RE POS
LINK_TYPE_SYMBOLIC	NUMBER	3	Type of link to be created	DBMS_XDB_RE POS
ON_DENY_NEXT_CUSTOM	NUMBER	1	If access denied, the next custom authorization is tried	DBMS_XDB_CO NFIG
ON_DENY_BASIC	NUMBER	2	If access denied, basic authentication is used	DBMS_XDB_CO NFIG

Summary of DBMS_XDB Subprograms

This table lists the DBMS_XDB subprograms and briefly describes them.

Table 223-2   DBMS_XDB Package Subprograms

Subprogram	Description
ACLCHECKPRIVILEGES Function	Checks access privileges granted to the current user by specified ACL document on a resource whose owner is specified by the 'owner' parameter.
APPENDRESOURCEMETADATA Procedure	Takes in user-defined metadata either as a REF to XMLTYPE or an XMLTYPE and adds it to the desired resource
CHANGEOWNER Procedure	Changes the owner of the resource/s to the specified owner.
CHANGEPRIVILEGES Function	Adds a specified ACE to a specified resource's ACL
CHECKPRIVILEGES Function	Checks access privileges granted to the current user on the specified resource
CREATEFOLDER Function	Creates a new folder resource in the hierarchy
CREATEOIDPATH Function	Creates a virtual path to the resource based on object ID
CREATERESOURCE Functions	Creates a new resource
DELETERESOURCE Procedure	Deletes a resource from the hierarchy
DELETERESOURCEMETADATA Procedures	Deletes metadata from a resource (can be used for schema-based or nonschema-based metadata)
ENABLEDIGESTAUTHENTICATION Procedure	Enables digest authentication
EXISTSRESOURCE Function	Determines if a resource is the hierarchy, based on its absolute path
GETACLDOCUMENT Function	Retrieves ACL document that protects resource given its path name
GETCONTENTBLOB Function	Retrieves the contents of a resource returned as a BLOB
GETCONTENTCLOB Function	Retrieves the contents of a resource returned as a CLOB
GETCONTENTVARCHAR2 Function	Retrieves the contents of a resource returned as a string

Table 223-2   (Cont.) DBMS_XDB Package Subprograms

Subprogram	Description
GETCONTENTXMLREF Function	Retrieves the contents of a resource returned as a a `REF` to an `XMLTYPE`
GETCONTENTXMLTYPE Function	Retrieves the contents of a resource returned as an `XMLTYPE`
GETHTTPREQUESTHEADER Function	Gets the values of the passed header
GETLOCKTOKEN Procedure	Returns that resource's lock token for the current user given a path to a resource
GETPRIVILEGES Function	Gets all privileges granted to the current user on a specified resource
GETRESOID Function	Returns the object ID of the resource from its absolute path
GETXDB_TABLESPACE Function	Returns the current tablespace of the XDB (user)
HASBLOBCONTENT Function	Returns `TRUE` if the resource has BLOB content
HASCHARCONTENT Function	Returns `TRUE` if the resource has character content
HASXMLCONTENT Function	Returns `TRUE` if the resource has XML content
HASXMLREFERENCE Function	Returns `TRUE` if the resource has `REF` to XML content
ISFOLDER Function	Returns `TRUE` if the resource is a folder or container
LINK Procedures	Creates a link to an existing resource
LOCKRESOURCE Function	Gets a WebDAV-style lock on that resource given a path to that resource
PROCESSLINKS Procedure	Processes document links in the specified resource
PURGERESOURCEMETADATA Procedure	Deletes all user metadata from a resource
RENAMERESOURCE Procedure	Renames the XDB resource
SETACL Procedure	Sets the ACL on a specified resource
SPLITPATH Procedure	Splits the path into a parentpath and childpath
TOUCHRESOURCE Procedure	Changes the modification time of the resource to the current time
UNLOCKRESOURCE Function	Unlocks the resource given a lock token and resource path
UPDATERESOURCEMETADATA Procedures	Updates metadata for a resource

ACLCHECKPRIVILEGES Function

This function checks access privileges granted to the current user by specified ACL document by the `OWNER` of the resource. Returns positive integer if all privileges are granted.

> **✎ Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the ACLCHECKPRIVILEGES Function.

Syntax

```
DBMS_XDB.ACLCHECKPRIVILEGES(
   acl_path  IN  VARCHAR2,
   owner     IN  VARCHAR2,
   privs     IN  xmltype)
 RETURN PLS_INTEGER;
```

Parameters

Table 223-3   *ACLCHECKPRIVILEGES Function Parameters*

Parameter	Description
acl_path	Absolute path in the Hierarchy for ACL document
owner	Resource owner name; the pseudo user "DAV:owner" is replaced by this user during ACL privilege resolution
privs	An XMLType instance of the privilege element specifying the requested set of access privileges. See description for CHECKPRIVILEGES Function.

APPENDRESOURCEMETADATA Procedure

This procedure takes in user-defined metadata either as a REF to XMLTYPE or an XMLTYPE and adds it to the desired resource.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the APPENDRESOURCEMETADATA Procedure.

Syntax

```
DBMS_XDB.APPENDRESOURCEMETADATA (
 abspath   IN  VARCHAR2,
 metadata  IN  XMLTYPE);

DBMS_XDB.APPENDRESOURCEMETADATA (
 abspath   IN  VARCHAR2,
 metadata  IN  REF SYS.XMLTYPE);
```

Parameters

Table 223-4   APPENDRESOURCEMETADATA Procedure

Parameter	Description
abspath	Absolute path of the resource
metadata	Metadata can be schema based or nonschema-based. Schema-based metadata is stored in its own table.

Usage Notes

*   In the case in which a `REF` is passed in, the procedure stores the `REF` in the resource, and the metadata is stored in a separate table. In this case you are responsible for populating the `RESID` column for the metadata table. Note that the `REF` passed in must be unique. In other words, there must not be a `REF` with the same value in the resource metadata, as this would violate uniqueness of properties. An error is thrown if users attempt to add a `REF` that already exists.

*   In the case where the XMLTYPE is passed in, the data is parsed to determine if it is schema-based or not and stored accordingly.

CHANGEOWNER Procedure

This procedure changes the owner of the resource/s to the specified owner. This procedure is deprecated in Release 12*c*.

> **✎ Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the CHANGEOWNER Procedure.

Syntax

```
DBMS_XDB.CHANGEOWNER(
    abspath   IN   VARCHAR2,
    owner     IN   VARCHAR2,
    recurse   IN   BOOLEAN := FALSE);
```

Parameters

Table 223-5   CHANGEOWNER Procedure Parameters

Parameter	Description
abspath	Absolute path of the resource
owner	New owner for the resource
recurse	If `TRUE`, recursively change owner of all resources in the folder tree

CHANGEPRIVILEGES Function

This function adds a specified ACE to a specified resource's ACL. This procedure is deprecated in Release 12*c*.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the CHANGEPRIVILEGES Function.

Syntax

```
DBMS_XDB.CHANGEPRIVILEGES(
   res_path   IN    VARCHAR2,
   ace        IN    xmltype)
 RETURN PLS_INTEGER;
```

Parameters

Table 223-6   CHANGEPRIVILEGES Function Parameters

Parameter	Description
res_path	Path name of the resource for which privileges need to be changed
ace	An XMLType instance of the `<ace>` element which specifies the `<principal>`, the operation `<grant>` and the list of privileges

Return Values

A positive integer if the ACL was successfully modified.

Usage Notes

If no ACE with the same principal and the same operation (`grant`/`deny`) already exists in the ACL, the new ACE is added at the end of the ACL.

CHECKPRIVILEGES Function

This function checks access privileges granted to the current user on the specified resource.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the CHECKPRIVILEGES Function.

Syntax

```
DBMS_XDB.CHECKPRIVILEGES(
   res_path   IN  VARCHAR2,
   privs      IN  xmltype)
 RETURN PLS_INTEGER;
```

Parameters

Table 223-7   CHECKPRIVILEGES Function Parameters

Parameter	Description
res_path	Absolute path in the Hierarchy for resource
privs	An XMLType instance of the privilege element specifying the requested set of access privileges

Return Values

A positive integer if all requested privileges granted.

CREATEFOLDER Function

This deprecated function creates a new folder resource in the hierarchy.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the CREATEFOLDER Function.

Syntax

```
DBMS_XDB.CREATEFOLDER(
   path   IN  VARCHAR2)
 RETURN BOOLEAN;
```

Parameters

Table 223-8   CREATEFOLDER Function Parameters

Parameter	Description
path	Path name for the new folder

Return Values

TRUE if operation successful; FALSE, otherwise.

Usage Notes

The given path name's parent folder must already exist in the hierarchy: if `'/folder1/folder2'` is passed as the path parameter, then `'/folder1'` must already exist.

CREATEOIDPATH Function

This deprecated function creates a virtual path to the resource based on object ID.

> **✎ Note:**
>
> This funtion is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the CREATEOIDPATH Function.

Syntax

```
DBMS_XDB.CREATEOIDPATH(
   oid    IN   RAW)
 RETURN VARCHAR2;
```

Parameters

Table 223-9   CREATEOIDPATH Function Parameters

Parameter	Description
oid	Object ID of the resource

CREATERESOURCE Functions

The deprecated function creates a new resource. The description of the overload options precede each version of the syntax

> **✎ Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the CREATERESOURCE Functions.

Syntax

Given a `REF` to an existing `XMLType` row, creates a resource whose contents point to that row. That row should not already exist inside another resource:

```
DBMS_XDB.CREATERESOURCE(
    abspath        IN  VARCHAR2,
    datarow        IN  REF SYS.XMLTYPE,
    createfolders  IN  BOOLEAN := FALSE)
 RETURN BOOLEAN;
```

Creates a resource with a specified `BLOB` as its contents, and specifies character set of the source `BLOB`:

```
DBMS_XDB.CREATERESOURCE(
    abspath        IN  VARCHAR2,
```

```
    data             IN  BLOB,
    csid             IN  NUMBER :=0,
    createfolders    IN  BOOLEAN := FALSE)
 RETURN BOOLEAN;
```

Creates a resource with a specified BFILE as its contents, and specifies character set of the source BFILE:

```
DBMS_XDB.CREATERESOURCE (
    abspath          IN  VARCHAR2,
    data             IN  BFILE,
    csid             IN  NUMBER :=0,
    createfolders    IN  BOOLEAN := FALSE)
 RETURN BOOLEAN;
```

Creates a resource with a specified CLOB as its contents:

```
DBMS_XDB.CREATERESOURCE (
    abspath          IN  VARCHAR2,
    data             IN  CLOB,
    createfolders    IN  BOOLEAN := FALSE)
 RETURN BOOLEAN;
```

Given a string, inserts a new resource into the hierarchy with the string as the contents:

```
DBMS_XDB.CREATERESOURCE (
    abspath          IN  VARCHAR2,
    data             IN  VARCHAR2,
    schemaurl        IN  VARCHAR2 := NULL,
    elem             IN  VARCHAR2 := NULL)
 RETURN BOOLEAN;
```

Given an XMLTYPE and a schema URL, inserts a new resource into the hierarchy with the XMLTYPE as the contents:

```
DBMS_XDB.CREATERESOURCE (
    abspath          IN  VARCHAR2,
    data             IN  SYS.XMLTYPE,
    schemaurl        IN  VARCHAR2 := NULL,
    elem             IN  VARCHAR2 := NULL)
 RETURN BOOLEAN;
```

Parameters

Table 223-10   CREATERESOURCE Function Parameters

Parameter	Description
abspath	Absolute path of the resource to create. The path name's parent folder must already exist in the hierarchy. In other words, if /foo/bar.txt is passed in, then folder /foo must already exist.
data	String buffer containing new resource's contents. The data is parsed to check if it contains a schema-based XML document, and the contents are stored as schema-based in the schema's default table. Otherwise, it is saved as binary data.
datarow	REF to an XMLType row to be used as the contents

Table 223-10   (Cont.) CREATERESOURCE Function Parameters

Parameter	Description
csid	Character set id of the document. Must be a valid Oracle ID; otherwise returns an error.
	If CSID is not specified, or if a zero CSID is specified, then the character set id of the document is determined as follows:
	• From the abspath extension, determine the resource's MIME type.
	• If the MIME type is */xml, then the encoding is detected based on Appendix F of the W3C XML 1.0 Reference at http://www.w3.org/TR/2000/REC-xml-20001006;
	• Otherwise, it is defaulted to the database character set.
createfolders	If TRUE, create the parent folders if they do not exist
schemaurl	For XML data, schema URL data conforms to (default NULL)
elem	Element name (default NULL)

Return Values

TRUE if operation successful; FALSE, otherwise.

DELETERESOURCE Procedure

This **deprecated procedure** deletes a resource from the hierarchy.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the DELETERESOURCE Procedure.

Syntax

```
DBMS_XDB.DELETERESOURCE(
   path           IN      VARCHAR2,
   delete_option IN      PLS_INTEGER);
```

Parameters

Table 223-11   DELETERESOURCE Procedure Parameters

Parameter	Description
path	Path name of the resource to delete
delete_option	The option that controls how a a resource is deleted; defined in Table 223-1:
	• DELETE_RESOURCE
	• DELETE_RECURSIVE
	• DELETE_FORCE
	• DELETE_RECURSIVE_FORCE

DELETERESOURCEMETADATA Procedures

This **deprecated procedure** takes in a resource by absolute path and removes either the schema-based metadata identified by the REF, or the metadata identified by the namespace and name combination, which can be either schema-based or non-schema based. It also takes an additional (optional) parameter that specifies how to delete it. This parameter is only relevant for schema-based resource metadata that needs to be deleted. For non-schema based metadata, this parameter is ignored.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the DELETERESOURCEMETADATA Procedures.

Syntax

Can be used only for schema-based metadata:

```
DBMS_XDB.DELETERESOURCEMETADATA (
   abspath        IN  VARCHAR2,
   metadata       IN  REF SYS.XMLTYPE,
   delete_option  IN  pls_integer := dbms_xdb.DELETE_RESOURCE_METADATA_CASCADE);
```

Can be used for schema-based or nonschema-based metadata:

```
DBMS_XDB.DELETERESOURCEMETADATA (
   abspath        IN VARCHAR2,
   metadatans     IN VARCHAR2,
   metadataname   IN VARCHAR2,
   delete_option  IN pls_integer := dbms_xdb.DELETE_RESOURCE_METADATA_CASCADE);
```

Parameters

Table 223-12   DELETERESOURCEMETADATA Procedure Parameters

Parameter	Description
abspath	Absolute path of the resource
metadata	REF to the piece of metadata (schema based) to be deleted
mettadatans	Namespace of the metadata fragment to be removed
mettadataname	Local name of the metadata fragment to be removed
delete_option	Only applicable for schema-based metadata, this can be one of the following: • DELETE_RES_METADATA_CASCADE - deletes the corresponding row in the metadata table • DELETE_RES_METADATA_NOCASCADE - does not delete the row in the metadata table

ENABLEDIGESTAUTHENTICATION Procedure

This deprecated procedure enabling digest authentication. It will list digest as the first authentication mechanism to be used by the XML DB HTTP server.

> **✎ Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_CONFIG package - the ENABLEDIGESTAUTHENTICATION Procedure.

Syntax

```
DBMS_XDB.ENABLEDIGESTAUTHENTICATION;
```

EXISTSRESOURCE Function

This deprecated function indicates if a resource is in the hierarchy. Matches resource by a string that represents its absolute path.

> **✎ Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the EXISTSRESOURCE Function.

Syntax

```
DBMS_XDB.EXISTSRESOURCE(
   abspath    IN    VARCHAR2)
 RETURN BOOLEAN;
```

Parameters

Table 223-13   EXISTSRESOURCE Function Parameters

Parameter	Description
abspath	Path name of the resource whose ACL document is required

Return Values

TRUE if the resource is found.

GETACLDOCUMENT Function

This deprecated function retrieves ACL document that protects resource given its path name.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the GETACLDOCUMENT Function.

Syntax

```
DBMS_XDB.GETACLDOCUMENT(
   abspath   IN   VARCHAR2)
 RETURN sys.xmltype;
```

Parameters

Table 223-14   GETACLDOCUMENT Function Parameters

Parameter	Description
abspath	Path name of the resource whose ACL document is required

Return Values

The `XMLType` for ACL document.

GETCONTENTBLOB Function

This deprecated function retrieves the contents of a resource returned as a `BLOB`.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the GETCONTENTBLOB Function.

Syntax

```
DBMS_XDB.GETCONTENTBLOB(
    abspath   IN     VARCHAR2,
    csid      OUT    PLS_INTEGER,
    locksrc   IN     BOOLEAN := FALSE)
  RETURN BLOB;
```

Parameters

Table 223-15   GETCONTENTBLOB Function Parameters

Parameter	Description
abspath	Absolute path of the resource
csid	If `TRUE`, lock and return the source LOB. If `FALSE`, return a temp LOB copy.
locksrc	Contents of the resource as a BLOB

Return Values

The contents of the resource as a BLOB.

GETCONTENTCLOB Function

This deprecated function gets the contents of a resource returned as a `CLOB`.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the GETCONTENTCLOB Function.

Syntax

```
DBMS_XDB.GETCONTENTCLOB(
    abspath   IN    VARCHAR2,
  RETURN CLOB;
```

Parameters

Table 223-16   GETCONTENTCLOB Function Parameters

Parameter	Description
abspath	Absolute path of the resource

Return Values

The contents of the resource as a CLOB.

GETCONTENTVARCHAR2 Function

This deprecated function gets the contents of a resource returned as a string.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the GETCONTENTVARCHAR2 Function.

Syntax

```
DBMS_XDB.GETCONTENTVARCHAR2(
     abspath    IN    VARCHAR2,
  RETURN BLOB;
```

Parameters

Table 223-17   GETCONTENTVARCHAR2 Function Parameters

Parameter	Description
abspath	Absolute path of the resource

Return Values

The contents of the resource as a string.

GETCONTENTXMLREF Function

This deprecated function retrieves the contents of a resource returned as a a `REF` to an `XMLTYPE`.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the GETCONTENTXMLREF Function.

Syntax

```
DBMS_XDB.GETCONTENTXMLREF(
     abspath    IN    VARCHAR2,
  RETURN SYS.XMLTYPE;
```

Parameters

Table 223-18   GETCONTENTXMLREF Function Parameters

Parameter	Description
abspath	Absolute path of the resource

Return Values

The contents of the resource as a `REF` to an `XMLTYPE`.

GETCONTENTXMLTYPE Function

This deprecated function retrieves the contents of a resource returned as an `XMLTYPE`.

> **✎ Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the GETCONTENTXMLTYPE Function.

Syntax

```
DBMS_XDB.GETCONTENTXMLTYPE(
     abspath    IN    VARCHAR2,
  RETURN SYS.XMLTYPE;
```

Parameters

Table 223-19   GETCONTENTXMLTYPE Function Parameters

Parameter	Description
abspath	Absolute path of the resource

Return Values

The contents of the resource as an XMLTYPE.

GETHTTPREQUESTHEADER Function

This deprecated function, if called during an HTTP request serviced by XDB, returns the values of the passed header. It is used by routines that implement custom authentication.

> **✎ Note:**
>
> This procedure is deprecated in Release 12*c*.

Syntax

```
DBMS_XDB.GETHTTPREQUESTHEADER(
    header_name    IN    VARCHAR2)
 RETURN VARCHAR2;
```

Parameters

Table 223-20   GETHTTPREQUESTHEADER Function Parameters

Parameter	Description
header_name	Passed header

Return Values

Returns NULL in case the header is not present in the request, or for AUTHENTICATION, for security reasons.

GETLOCKTOKEN Procedure

Given a path to a resource, this deprecated procedure returns that resource's lock token for the current user.

> **✎ Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the GETLOCKTOKEN Procedure.

Syntax

```
DBMS_XDB.GETLOCKTOKEN(
    path        IN     VARCHAR2,
    locktoken   OUT    VARCHAR2);
```

Parameters

Table 223-21   GETLOCKTOKEN Procedure Parameters

Parameter	Description
path	Path name to the resource
locktoken	Logged-in user's lock token for the resource

Usage Notes

The user must have READPROPERTIES privilege on the resource.

GETPRIVILEGES Function

This deprecated function gets all privileges granted to the current user on a specified resource.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the GETPRIVILEGES Function.

Syntax

```
DBMS_XDB.GETPRIVILEGES(
   res_path    IN     VARCHAR2)
 RETURN sys.xmltype;
```

Parameters

Table 223-22   GETPRIVILEGES Function Parameters

Parameter	Description
res_path	Absolute path in the hierarchy of the resource

Return Values

An `XMLType` instance of `<privilege>` element, which contains the list of all leaf privileges granted on this resource to the current user.

GETRESOID Function

This deprecated procedure returns the object ID of the resource from its absolute path.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the GETRESOID Function.

Syntax

```
DBMS_XDB.GETRESOID(
   abspath IN VARCHAR2)
RETURN RAW;
```

ORACLE

Parameters

Table 223-23   GETRESOID Function Parameters

Parameter	Description
abspath_path	Absolute path of the resource

Return Values

NULL if the resource is not present.

GETXDB_TABLESPACE Function

This deprecated function returns the current tablespace of the XDB (user).

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the GETXDB_TABLESPACE Function.

Syntax

```
DBMS_XDB.GETXDB_TABLESPACE
 RETURN VARCHAR2;
```

HASBLOBCONTENT Function

This deprecated function returns TRUE if the resource has BLOB content.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the HASBLOBCONTENT Function.

Syntax

```
DBMS_XDB.HASBLOBCONTENT
    abspath   IN     VARCHAR2)
 RETURN BOOLEAN;
```

Parameters

Table 223-24   HASBLOBCONTENT Function Parameters

Parameter	Description
abspath_path	Absolute path of the resource

Return Values

`TRUE` if the resource has BOB content.

HASCHARCONTENT Function

This deprecated function returns `TRUE` if the resource has character content.

> **✎ Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the HASCHARCONTENT Function.

Syntax

```
DBMS_XDB.HASCHARCONTENT
    abspath    IN    VARCHAR2)
 RETURN BOOLEAN;
```

Parameters

Table 223-25   HASCHARCONTENT Function Parameters

Parameter	Description
abspath_path	Absolute path of the resource

Return Values

`TRUE` if the resource has character content.

HASXMLCONTENT Function

This deprecated function returns `TRUE` if the resource has XML content.

> **✎ Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the HASXMLCONTENT Function.

Syntax

```
DBMS_XDB.HASXMLCONTENT
    abspath    IN    VARCHAR2)
 RETURN BOOLEAN;
```

Parameters

Table 223-26   HASXMLCONTENT Function Parameters

Parameter	Description
abspath_path	Absolute path of the resource

Return Values

TRUE if the resource has XML content.

HASXMLREFERENCE Function

This deprecated function returns TRUE if the resource has a REF to XML content.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the HASXMLREFERENCE Function.

Syntax

```
DBMS_XDB.HASXMLREFERENCE
     abspath    IN     VARCHAR2)
 RETURN BOOLEAN;
```

Parameters

Table 223-27   HASXMLREFERENCE Function Parameters

Parameter	Description
abspath_path	Absolute path of the resource

Return Values

TRUE resource has a REF to XML content.

ISFOLDER Function

This deprecated function returns TRUE if the resource is a folder or container.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the ISFOLDER Function.

Syntax

```
DBMS_XDB.ISFOLDER
     abspath   IN    VARCHAR2)
 RETURN BOOLEAN;
```

Parameters

Table 223-28   DBMS_XDB.ISFOLDER Function Parameters

Parameter	Description
abspath_path	Absolute path of the resource

Return Values

TRUE if the resource is a folder or container.

LINK Procedures

This deprecated procedure creates a link from a specified folder to a specified resource.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the LINK Procedures.

Syntax

```
DBMS_XDB.LINK(
   srcpath      IN   VARCHAR2,
   linkfolder   IN   VARCHAR2,
   linkname     IN   VARCHAR2);

DBMS_XDB.LINK(
   srcpath      IN   VARCHAR2,
   linkfolder   IN   VARCHAR2,
   linkname     IN   VARCHAR2,
   linktype     IN   PLS_INTEGER := DBMS_XDB.LINK_TYPE_HARD);
```

Parameters

Table 223-29   LINK Procedure Parameters

Parameter	Description
srcpath	Path name of the resource to which a link is created
linkfolder	Folder in which the new link is placed
linkname	Name of the new link

Table 223-29   (Cont.) LINK Procedure Parameters

Parameter	Description
linktype	Type of link to be created:
	• DBMS_XDB.LINK_TYPE_HARD (default)
	• DBMS_XDB.LINK_TYPE_WEAK
	• DBMS_XDB.LINK_TYPE_SYMBOLIC

LOCKRESOURCE Function

Given a path to a resource, this deprecated function gets a WebDAV-style lock on that resource.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the LOCKRESOURCE Function.

Syntax

```
DBMS_XDB.LOCKRESOURCE(
   path      IN  VARCHAR2,
   depthzero IN  BOOLEAN,
   shared    IN  boolean)
RETURN BOOLEAN;
```

Parameters

Table 223-30   LOCKRESOURCE Function Parameters

Parameter	Description
path	Path name of the resource to lock.
depthzero	Currently not supported
shared	Passing TRUE obtains a shared write lock

Return Values

TRUE if successful.

Usage Notes

The user must have UPDATE privileges on the resource.

PROCESSLINKS Procedure

This deprecated procedure processes document links in the specified resource.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the PROCESSLINKS Procedure.

Syntax

```
DBMS_XDB.PURGERESOURCEMETADATA(
 abspath  IN  VARCHAR2,
 recurse  IN  BOOLEAN := FALSE);
```

Parameters

Table 223-31   PROCESSLINKS Procedure Parameters

Parameter	Description
abspath	Absolute path of the resource. If the path is a folder, use the recurse flag.
recurse	Used only if abspath specifies a folder. If TRUE, process links of all resources in the folder hierarchy rooted at the specified resource. If FALSE, process links of all documents in this folder only.

PURGERESOURCEMETADATA Procedure

This deprecated procedure deletes all user metadata from a resource. Schema-based metadata is removed in cascade mode, rows being deleted from the corresponding metadata tables.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the PURGERESOURCEMETADATA Procedure.

Syntax

```
DBMS_XDB.PURGERESOURCEMETADATA(
 abspath  IN  VARCHAR2);
```

Parameters

Table 223-32   PURGERESOURCEMETADATA Procedure Parameters

Parameter	Description
abspath	Absolute path of the resource

RENAMERESOURCE Procedure

This deprecated procedure renames the XDB resource.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the RENAMERESOURCE Procedure.

Syntax

```
DBMS_XDB.RENAMERESOURCE(
   srcpath    IN  VARCHAR2,
   destfolder IN  CARCHAR2,
   newname    IN  VARCHAR2);
```

Parameters

Table 223-33   RENAMERESOURCE Procedure Parameters

Parameter	Description
srcpath	Absolute path in the Hierarchy for the source resource destination folder
destfolder	Absolute path in the Hierarchy for the destination folder
newname	Name of the child in the destination folder

SETACL Procedure

This deprecated procedure set the ACL on a specified resource to be the ACL specified by path.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the SETACL Procedure.

Syntax

```
DBMS_XDB.SETACL(
   res_path   IN  VARCHAR2,
   acl_path   IN  VARCHAR2);
```

Parameters

Table 223-34   SETACL Procedure Parameters

Parameter	Description
res_path	Absolute path in the Hierarchy for resource
acl_path	Absolute path in the Hierarchy for ACL

Usage Notes

The user must have `<write-acl>` privileges on the resource.

SPLITPATH Procedure

This deprecated procedure splits the path into a parentpath and childpath.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the SPLITPATH Procedure.

Syntax

```
DBMS_XDB.SPLITPATH(
      abspath      IN  VARCHAR2,
     parentpath  OUT VARCHAR2,
    childpath     OUT VARCHAR2);
```

Parameters

Table 223-35   SPLITPATH Procedure Parameters

Parameter	Description
abspath	Absolute path to be split
parentpath	Parentpath
childpath	Childpath

TOUCHRESOURCE Procedure

This **deprecated procedure** changes the modification time of the resource to the current time.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the TOUCHRESOURCE Procedure.

Syntax

```
DBMS_XDB.TOUCHRESOURCE
     abspath   IN    VARCHAR2);
```

Parameters

Table 223-36   DBMS_XDB.TOUCHRESOURCE Procedure Parameters

Parameter	Description
abspath_path	Absolute path of the resource

UNLOCKRESOURCE Function

This deprecated function unlocks the resource given a lock token and a path to the resource.

> **Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the UNLOCKRESOURCE Function.

Syntax

```
DBMS_XDB.UNLOCKRESOURCE(
   path     IN  VARCHAR2,
   deltoken IN  VARCHAR2)
 RETURN BOOLEAN;
```

Parameters

Table 223-37   UNLOCKRESOURCE Function Parameters

Parameter	Description
path	Path name to the resource
deltoken	Lock token to be removed

Return Values

`TRUE` if operation successful.

Usage Notes

The user must have `UPDATE` privileges on the resource.

UPDATERESOURCEMETADATA Procedures

This deprecated procedure updates metadata for a resource.

> **✎ Note:**
>
> This procedure is deprecated in Release 12*c*. This functionality is replaced by a subprogram of the same name in the DBMS_XDB_REPOS package - the UPDATERESOURCEMETADATA Procedures.

The procedure takes in a resource identified by absolute path and the metadata in it to replace identified by its `REF`. It replaces that piece of metadata with user-defined metadata which is either in the form of a `REF` to `XMLTYPE` or an `XMLTYPE`.

Syntax

Can be used to update schema-based metadata only. The new metadata must be schema-based:

```
DBMS_XDB.UPDATERESOURCEMETADATA(
   abspath   IN VARCHAR2,
   oldmetadata    IN    REF SYS.XMLTYPE,
   newmetadata    IN    REF SYS.XMLTYPE)
```

Can be used to update schema-based metadata only. The new metadata must be schema-based or nonschema-based:

```
DBMS_XDB.UPDATERESOURCEMETADATA(
   abspath        IN   VARCHAR2,
   oldmetadata    IN   REF SYS.XMLTYPE,
   newmetadata    IN   XMLTYPE);
```

Can be used for both schema-based and nonschema-based metadata:

```
DBMS_XDB.UPDATERESOURCEMETADATA(
   abspath        IN   VARCHAR2,
   oldns          IN   VARCHAR2,
   oldname IN VARCHAR,
   newmetadata IN XMLTYPE);
```

Can be used for both schema-based or nonschema-based metadata. New metadata must be schema-based:

```
DBMS_XDB.UPDATERESOURCEMETADATA(
   abspath        IN    VARCHAR2,
```

```
oldns        IN   VARCHAR2,
oldname      IN   VARCHAR,
newmetadata  IN   REF SYS.XMLTYPE);
```

Parameters

Table 223-38   UPDATERESOURCEMETADATA Procedure Parameters

Parameter	Description
abspath	Absolute path of the resource
oldmetadata	REF to the old of metadata
newmetadata	REF to the new, replacement metadata (can be either schema-based or nonschema-based depending on the overload)
oldns	Namespace identifying old metadata
oldname	Local name identifying old metadata

Usage Notes

In the case of REF, it stores the REF in the resource and the metadata is stored in a separate table. Uniqueness of REFs is enforced. In the case where the XMLTYPE is passed in, data is parsed to determine if it is schema-based or not and is stored accordingly.