26

PL/SQL Access to Oracle XML DB Repository

PL/SQL packages DBMS_XDB_CONFIG and DBMS_XDB_REPOS together provide the Oracle XML DB resource application program interface (API) for PL/SQL. You use the former to configure Oracle XML DB and its repository. You use the latter to perform other, non-configuration operations on the repository.

> **Note:**
>
> The Oracle XML DB Repository is deprecated with Oracle Database 23ai.

- DBMS_XDB_REPOS: Access and Manage Repository Resources
  You use PL/SQL package DBMS_XDB_REPOS to access and manage resources in Oracle XML DB Repository. It includes methods for managing resource security based on access control lists (ACLs). An ACL is a list of access control entries (ACEs) that determines which principals (users and roles) have access to which resources.
- DBMS_XDB_REPOS: ACL-Based Security Management
  PL/SQL package DBMS_XDB_REPOS provides Oracle XML DB ACL-based security management functions and procedures.
- DBMS_XDB_CONFIG: Configuration Management
  PL/SQL package DBMS_XDB_CONFIG provides Oracle XML DB configuration management functions and procedures.

Related Topics

- PL/SQL APIs for XMLType: References
  The PL/SQL Application Programming Interfaces (APIs) for XMLType are described.
- Package DBMS_XDB_ADMIN
  PL/SQL package DBMS_XDB_ADMIN has subprograms for managing and configuring Oracle XML DB and Oracle XML DB Repository.

DBMS_XDB_REPOS: Access and Manage Repository Resources

You use PL/SQL package DBMS_XDB_REPOS to access and manage resources in Oracle XML DB Repository. It includes methods for managing resource security based on access control lists (ACLs). An ACL is a list of access control entries (ACEs) that determines which principals (users and roles) have access to which resources.

> **See Also:**
>
> *Oracle Database PL/SQL Packages and Types Reference*

Table 26-1 describes the functions and procedures in package DBMS_XDB_REPOS.

Table 26-1 DBMS_XDB_REPOS Resource Access and Management Subprograms

Function/Procedure	Description
addResource	Insert a new file resource into the repository hierarchy, with the given string as its contents.
appendResourceMetadata	Add user-defined metadata to a resource.
createFolder	Create a new folder resource.
createOIDPath	Create a virtual path to a resource, based on its object identifier (OID).
createResource	Create a new file resource.
deleteResource	Delete a resource from the repository.
deleteResourceMetadata	Delete specific user-defined metadata from a resource.
existsResource	Indicate whether or not a resource exists, given its absolute path.
getContentBLOB	Return the contents of a resource as a BLOB instance.
getContentVARCHAR2	Return the contents of a resource as a VARCHAR2 value.
getContentXMLRef	Return the contents of a resource as a reference to an XMLType instance.
getContentXMLType	Return the contents of a resource as an XMLType instance.
getLockToken	Return a resource lock token for the current user, given a path to the resource.
getResOID	Return the object identifier (OID) of a resource, given its absolute path.
getResource	Return the instance of class DBMS_XDBRESOURCE.XDBResource that is located at a given path in the repository.
getXDB_tablespace	Return the current tablespace of database schema (user account) XDB.
hasBLOBContent	Return TRUE if a given resource has BLOB content.
hasCharContent	Return TRUE if a given resource has character content.
hasXMLContent	Return TRUE if a given resource has XMLType content.
hasXMLReference	Return TRUE if a given resource has a reference to XMLType content.
isFolder	Return TRUE if a given resource is a folder.
link	Create a link to an existing resource.
lockResource	Obtain a WebDAV-style lock on a resource, given a path to the resource.
processLinks	Process all XLink (*deprecated*) and XInclude links in a document or folder.
purgeResourceMetadata	Delete all user-defined metadata from a given resource.
refreshContentSize	Recompute the content size of a given resource.
renameResource	Rename a given resource.
touchResource	Change the last-modified time to the current time.
unlockResource	Unlock a resource, given its lock token and path.
updateResourceMetadata	Modify user-defined resource metadata.

Example 26-1 uses package DBMS_XDB_REPOS to manage repository resources. It creates the following:

- A folder, mydocs, under folder /public
- Two file resources, emp_selby.xml and emp_david.xml

- Two links to the file resources `person_selby.xml` and `person_david.xml`

It then deletes each of the newly created resources and links. The folder contents are deleted before the folder itself.

> **See Also:**
>
> User-Defined Repository Metadata for examples using `appendResourceMetadata` and `deleteResourceMetadata`

Example 26-1 Managing Resources Using DBMS_XDB_REPOS

```
DECLARE
  retb BOOLEAN;
BEGIN
  retb := DBMS_XDB_REPOS.createfolder('/public/mydocs');
  retb := DBMS_XDB_REPOS.createresource('/public/mydocs/emp_selby.xml',
                                        '<emp_name>selby</emp_name>');
  retb := DBMS_XDB_REPOS.createresource('/public/mydocs/emp_david.xml',
                                        '<emp_name>david</emp_name>');
END;
/
PL/SQL procedure successfully completed.

CALL DBMS_XDB_REPOS.link('/public/mydocs/emp_selby.xml',
                         '/public/mydocs',
                         'person_selby.xml');
Call completed.

CALL DBMS_XDB_REPOS.link('/public/mydocs/emp_david.xml',
                         '/public/mydocs',
                         'person_david.xml');
Call completed.

CALL DBMS_XDB_REPOS.deleteresource('/public/mydocs/emp_selby.xml');
Call completed.

CALL DBMS_XDB_REPOS.deleteresource('/public/mydocs/person_selby.xml');
Call completed.

CALL DBMS_XDB_REPOS.deleteresource('/public/mydocs/emp_david.xml');
Call completed.

CALL DBMS_XDB_REPOS.deleteresource('/public/mydocs/person_david.xml');
Call completed.

CALL DBMS_XDB_REPOS.deleteresource('/public/mydocs');
Call completed.
```

DBMS_XDB_REPOS: ACL-Based Security Management

PL/SQL package `DBMS_XDB_REPOS` provides Oracle XML DB ACL-based security management functions and procedures.

Table 26-2 DBMS_XDB_REPOS: Security Management Subprograms

Function/Procedure	Description
`ACLCheckPrivileges`	Check the access privileges granted to the current user by an ACL.
`changeOwner`	Change the owner of a given resource to a given user.
`changePrivileges`	Add an ACE to a resource ACL.
`checkPrivileges`	Check the access privileges granted to the current user for a resource.
`getACLDocument`	Return the ACL document that protects a resource, given the path name of the resource.
`getPrivileges`	Return all privileges granted to the current user for a resource.
`setACL`	Set the ACL for a resource.

> **See Also:**
>
> *Oracle Database PL/SQL Packages and Types Reference*

In Example 26-2, database user `HR` creates two resources: a folder, `/public/mydocs`, with a file in it, `emp_selby.xml`. Procedure `getACLDocument` is called on the file resource, showing that the `<principal>` user for the document is `PUBLIC`.

In Example 26-3, the system manager connects and uses procedure `setACL` to give the owner (database schema `HR`) all privileges on the file resource created in Example 26-2. Procedure `getACLDocument` then shows that the `<principal>` user is `dav:owner`, the owner (`HR`).

In Example 26-4, user `HR` connects and uses function `changePrivileges` to add a new access control entry (ACE) to the ACL, which gives all privileges on resource `emp_selby.xml` to user `oe`. Procedure `getACLDocument` shows that the new ACE was added to the ACL.

In Example 26-5, user `oe` connects and calls `DBMS_XDB_REPOS.getPrivileges`, which shows all of the privileges granted to user `oe` on resource `emp_selby.xml`.

Example 26-2 Using DBMS_XDB_REPOS.GETACLDOCUMENT

```
CONNECT hr
Enter password: password

Connected.

DECLARE
  retb BOOLEAN;
BEGIN
  retb := DBMS_XDB_REPOS.createFolder('/public/mydocs');
  retb := DBMS_XDB_REPOS.createResource('/public/mydocs/emp_selby.xml',
                                        '<emp_name>selby</emp_name>');
END;
/
PL/SQL procedure successfully completed.

SELECT XMLSerialize(DOCUMENT
                    DBMS_XDB_REPOS.getACLDocument('/public/mydocs/emp_selby.xml')
                    AS CLOB)
  FROM DUAL;

XMLSERIALIZE(DOCUMENTDBMS_XDB_REPOS.GETACLDOCUMENT('/PUBLIC/MYDOCS/EMP_SELBY.XML
--------------------------------------------------------------------------------
<acl description="Public:All privileges to PUBLIC" xmlns="http://xmlns.oracle.co
m/xdb/acl.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaL
ocation="http://xmlns.oracle.com/xdb/acl.xsd                           http://xm
lns.oracle.com/xdb/acl.xsd" shared="true">
  <ace>
    <grant>true</grant>
    <principal>PUBLIC</principal>
    <privilege>
      <all/>
    </privilege>
  </ace>
</acl>

1 row selected.
```

Example 26-3 Using DBMS_XDB_REPOS.SETACL

```
CONNECT SYSTEM
Enter password: password

Connected.

-- Give all privileges to owner, HR.
CALL DBMS_XDB_REPOS.setACL('/public/mydocs/emp_selby.xml',
                           '/sys/acls/all_owner_acl.xml');
Call completed.
COMMIT;
Commit complete.

SELECT XMLSerialize(DOCUMENT
                    DBMS_XDB_REPOS.getACLDocument('/public/mydocs/emp_selby.xml')
                    AS CLOB)
  FROM DUAL;

XMLSERIALIZE(DOCUMENTDBMS_XDB_REPOS.GETACLDOCUMENT('/PUBLIC/MYDOCS/EMP_SELBY.XML
--------------------------------------------------------------------------------
<acl description="Private:All privileges to OWNER only and not accessible to oth
ers" xmlns="http://xmlns.oracle.com/xdb/acl.xsd" xmlns:dav="DAV:" xmlns:xsi="htt
p://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.oracle.
com/xdb/acl.xsd http://xmlns.oracle.com/xdb/acl.xsd" shared="true">
  <ace>
    <grant>true</grant>
    <principal>dav:owner</principal>
    <privilege>
      <all/>
    </privilege>
  </ace>
</acl>

1 row selected.
```

Example 26-4 Using DBMS_XDB_REPOS.CHANGEPRIVILEGES

```
CONNECT hr
Enter password: password

Connected.

SET SERVEROUTPUT ON

-- Add an ACE giving privileges to user OE
DECLARE
  r        PLS_INTEGER;
  ace      XMLType;
  ace_data VARCHAR2(2000);
BEGIN
  ace_data := '<ace xmlns="http://xmlns.oracle.com/xdb/acl.xsd"
```

```
                    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                    xsi:schemaLocation="http://xmlns.oracle.com/xdb/acl.xsd
                                        http://xmlns.oracle.com/xdb/acl.xsd
                                        DAV:http://xmlns.oracle.com/xdb/dav.xsd">
                 <principal>OE</principal>
                 <grant>true</grant>
                 <privilege><all/></privilege>
               </ace>';
  ace := XMLType.createXML(ace_data);
  r := DBMS_XDB_REPOS.changePrivileges('/public/mydocs/emp_selby.xml', ace);
END;
/

PL/SQL procedure successfully completed.

COMMIT;

SELECT XMLSerialize(DOCUMENT
                    DBMS_XDB_REPOS.getACLDocument('/public/mydocs/emp_selby.xml')
                    AS CLOB)
  FROM DUAL;

XMLSERIALIZE(DOCUMENTDBMS_XDB_REPOS.GETACLDOCUMENT('/PUBLIC/MYDOCS/EMP_SELBY.XML
--------------------------------------------------------------------------------
<acl description="Private:All privileges to OWNER only and not accessible to oth
ers" xmlns="http://xmlns.oracle.com/xdb/acl.xsd" xmlns:dav="DAV:" xmlns:xsi="htt
p://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.oracle.
com/xdb/acl.xsd                           http://xmlns.oracle.com/xdb/acl.xsd" s
hared="false">
  <ace>
    <grant>true</grant>
    <principal>dav:owner</principal>
    <privilege>
      <all/>
    </privilege>
  </ace>
  <ace>
    <grant>true</grant>
    <principal>OE</principal>
    <privilege>
      <all/>
    </privilege>
  </ace>
</acl>

1 row selected.
```

Example 26-5   Using DBMS_XDB_REPOS.GETPRIVILEGES

```
CONNECT oe
Enter password: password

Connected.

SELECT XMLSerialize(DOCUMENT
                    DBMS_XDB_REPOS.getPrivileges('/public/mydocs/emp_selby.xml')
                    AS CLOB)
  FROM DUAL;

XMLSERIALIZE(DOCUMENTDBMS_XDB_REPOS.GETPRIVILEGES('/PUBLIC/MYDOCS/EMP_SELBY.XML'
--------------------------------------------------------------------------------
<privilege xmlns="http://xmlns.oracle.com/xdb/acl.xsd" xmlns:xsi="http://www.w3.
org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.oracle.com/xdb/acl
.xsd http://xmlns.oracle.com/xdb/acl.xsd DAV: http://xmlns.oracle.com/xdb/dav.xs
d" xmlns:xdbacl="http://xmlns.oracle.com/xdb/acl.xsd" xmlns:dav="DAV:">
  <read-properties/>
  <read-contents/>
  <write-config/>
  <link/>
  <unlink/>
  <read-acl/>
  <write-acl-ref/>
  <update-acl/>
  <resolve/>
  <link-to/>
  <unlink-from/>
  <dav:lock/>
  <dav:unlock/>
  <dav:write-properties/>
  <dav:write-content/>
  <dav:execute/>
  <dav:take-ownership/>
  <dav:read-current-user-privilege-set/>
</privilege>

1 row selected.
```

DBMS_XDB_CONFIG: Configuration Management

PL/SQL package `DBMS_XDB_CONFIG` provides Oracle XML DB configuration management functions and procedures.

> **Note:**
>
> Oracle recommends that you use the subprograms of PL/SQL package `DBMS_XDB_CONFIG` to set or change FTP or HTTP port numbers. Do *not* set ports by directly editing configuration file `xdbconfig.xml`

Table 26-3 DBMS_XDB_CONFIG: Configuration Management Subprograms

Subprogram	Description
`addHTTPExpireMapping`	Add a mapping of a URL pattern to an expiration date to table `XDB$CONFIG`. The mapping controls the `Expire` headers for URLs that match the pattern.
`addMIMEMapping`	Add a MIME mapping to table `XDB$CONFIG`.
`addSchemaLocMapping`	Add a schema-location mapping to table `XDB$CONFIG`.
`addServlet`	Add a servlet to table `XDB$CONFIG`.
`addServletMapping`	Add a servlet mapping to table `XDB$CONFIG`.
`addServletSecRole`	Add a security role reference to a servlet.
`addXMLExtension`	Add an XML extension to table `XDB$CONFIG`.
`cfg_get`	Return the configuration information for the current session.
`cfg_refresh`	Refresh the session configuration information using the current Oracle XML DB configuration file, `xdbconfig.xml`.
`cfg_update`	Update the Oracle XML DB configuration information. This writes the configuration file, `xdbconfig.xml`.
`deleteHTTPExpireMapping`	Delete *all* mappings of a given URL pattern to an expiration date from table `XDB$CONFIG`.
`deleteMIMEMapping`	Delete a MIME mapping from table `XDB$CONFIG`.
`deleteSchemaLocMapping`	Delete a schema-location mapping from table `XDB$CONFIG`.

Table 26-3 (Cont.) DBMS_XDB_CONFIG: Configuration Management Subprograms

Subprogram	Description
`deleteServlet`	Delete a servlet from table `XDB$CONFIG`.
`deleteServletMapping`	Delete a servlet mapping from table `XDB$CONFIG`.
`deleteServletSecRole`	Delete a security role reference from a servlet.
`deleteXMLExtension`	Delete an XML extension from table `XDB$CONFIG`.
`enableDigestAuthentication`	Enable digest authentication.
`getFTPPort`	Return the current FTP port number.
`getHTTPConfigRealm`	Return the HTTP configuration realm.
`getHTTPPort`, `getHTTPSPort`	Return the current HTTP(S) port number.
`getListenerEndPoint`	Return the parameters of a listener end point for the HTTP server.
`GetRemoteHTTPPort`, `GetRemoteHTTPSPort`	Return the number of the current remote HTTP(S) port.
`setFTPPort`	Set the Oracle XML DB FTP port to the specified port number.
`setHTTPConfigRealm`	Set the HTTP configuration realm.
`setHTTPPort`, `setHTTPSPort`	Set the Oracle XML DB HTTP(S) port to the specified port number.
`setListenerEndPoint`	Set the parameters of a listener end point for the HTTP server.
`setListenerLocalAccess`	Either (a) restrict all listener end points to listen on only the `localhost` interface or (b) allow all listener end points to listen on both `localhost` and non-`localhost` interfaces.
`SetRemoteHTTPPort`, `SetRemoteHTTPSPort`	Define the port number of a remote HTTP or HTTPS port, respectively. A remote port number is stored similarly to a non-remote HTTP(S) port, except they it is not specified in the configuration file, `xdbconfig.xml` file or in the configuration XML schema, `xdbconfig.xsd`.
`usedPort`	Return the ports used by other pluggable databases (PDBs) in the same multitenant container database (CDB). The return value is an `XMLType` instance that lists each PDB by id number and its associated ports by type and number.

> **See Also:**
>
> *Oracle Database PL/SQL Packages and Types Reference*, Chapter "DBMS_XDB_CONFIG"

Example 26-6 uses function `cfg_get` to retrieve the Oracle XML DB configuration file, `xdbconfig.xml`.

Example 26-7 illustrates the use of procedure `cfg_update`. The current configuration is retrieved as an `XMLType` instance and modified. It is then rewritten using `cfg_update`.

Example 26-6 Using DBMS_XDB_CONFIG.CFG_GET

```
CONNECT SYSTEM
Enter password: password

Connected.
```

```
SELECT DBMS_XDB_CONFIG.cfg_get() FROM DUAL;

DBMS_XDB_CONFIG.CFG_GET()
--------------------------------------------------------------------------------
<xdbconfig xmlns="http://xmlns.oracle.com/xdb/xdbconfig.xsd" xmlns:xsi="http://w
ww.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.oracle.com/x
db/xdbconfig.xsd                                     http://xmlns.oracle.com/xdb
/xdbconfig.xsd">
  <sysconfig>
    <acl-max-age>19</acl-max-age>
    <acl-cache-size>32</acl-cache-size>
    <invalid-pathname-chars/>
    <case-sensitive>true</case-sensitive>
    <call-timeout>6000</call-timeout>
    <max-link-queue>65536</max-link-queue>
    <max-session-use>100</max-session-use>
    <persistent-sessions>false</persistent-sessions>
    <default-lock-timeout>3600</default-lock-timeout>
    <xdbcore-logfile-path>/sys/log/xdblog.xml</xdbcore-logfile-path>
    <xdbcore-log-level>0</xdbcore-log-level>
    <resource-view-cache-size>1048576</resource-view-cache-size>
    <protocolconfig>
      <common>
        . . .
      </common>
      <ftpconfig>
        . . .
      </ftpconfig>
      <httpconfig>
        <http-port>0</http-port>
        <http-listener>local_listener</http-listener>
        <http-protocol>tcp</http-protocol>
        <max-http-headers>64</max-http-headers>
        <max-header-size>16384</max-header-size>
        <max-request-body>2000000000</max-request-body>
        <session-timeout>6000</session-timeout>
        <server-name>XDB HTTP Server</server-name>
        <logfile-path>/sys/log/httplog.xml</logfile-path>
        <log-level>0</log-level>
        <servlet-realm>Basic realm=&quot;XDB&quot;</servlet-realm>
        <webappconfig>
        . . .
        </webappconfig>
        <authentication>
        . . .
        </authentication>
    </protocolconfig>
    <xdbcore-xobmem-bound>1024</xdbcore-xobmem-bound>
    <xdbcore-loadableunit-size>16</xdbcore-loadableunit-size>
    <acl-evaluation-method>ace-order</acl-evaluation-method>
  </sysconfig>
</xdbconfig>

1 row selected.
```

Example 26-7   Using DBMS_XDB_CONFIG.CFG_UPDATE

```
DECLARE
  configxml     SYS.XMLType;
  configxml2    SYS.XMLType;
BEGIN
  -- Get the current configuration
  configxml := DBMS_XDB_CONFIG.cfg_get();

  -- Modify the configuration
SELECT XMLQuery(
  'declare default element namespace
   "http://xmlns.oracle.com/xdb/xdbconfig.xsd"; (: :)
   copy $i := $p1 modify
     (for $j in $i/xdbconfig/sysconfig/protocolconfig/httpconfig/http-port
      return replace value of node $j with $p2)
     return $i'
    PASSING CONFIGXML AS "p1", '8000' AS "p2" RETURNING CONTENT)
  INTO configxml2 FROM DUAL;

  -- Update the configuration to use the modified version
  DBMS_XDB_CONFIG.cfg_update(configxml2);
END;
/

PL/SQL procedure successfully completed.

SELECT DBMS_XDB_CONFIG.cfg_get() FROM DUAL;

DBMS_XDB_CONFIG.CFG_GET()
--------------------------------------------------------------------------------
<xdbconfig xmlns="http://xmlns.oracle.com/xdb/xdbconfig.xsd" xmlns:xsi="http://w
ww.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.oracle.com/x
db/xdbconfig.xsd http://xmlns.oracle.com/xdb/xdbconfig.xsd">
  <sysconfig>
    <acl-max-age>15</acl-max-age>
    <acl-cache-size>32</acl-cache-size>
    <invalid-pathname-chars/>
    <case-sensitive>true</case-sensitive>
    <call-timeout>6000</call-timeout>
    <max-link-queue>65536</max-link-queue>
    <max-session-use>100</max-session-use>
    <persistent-sessions>false</persistent-sessions>
    <default-lock-timeout>3600</default-lock-timeout>
    <xdbcore-logfile-path>/sys/log/xdblog.xml</xdbcore-logfile-path>
    <resource-view-cache-size>1048576</resource-view-cache-size>
    <protocolconfig>
      <common>
      . . .
      </common>
      <ftpconfig>
      . . .
      </ftpconfig>
      <httpconfig>
        <http-port>8000</http-port>
        . . .
      </httpconfig>
    </protocolconfig>
    <xdbcore-xobmem-bound>1024</xdbcore-xobmem-bound>
    <xdbcore-loadableunit-size>16</xdbcore-loadableunit-size>
    <acl-evaluation-method>ace-order</acl-evaluation-method>
</xdbconfig>

1 row selected.
```