
OZ Java Server

API Developer's Guide



Table of Contents

Part I. OZ Java Server API	1
1. Class	2
Cache	3
ConnectionPool	4
ConnectionPoolInfo	6
ConnectionPoolStatus	13
DataBind	14
HCDataModule	15
HCDataSet	17
Log	17
Mail	19
SortProperties	22
MaxRowsOfSet	23
MemoryStatus	24
Module	25
Monitor	30
OZCPException	32
OZDataSetInfo	32
OZODlgItem	33
OZParameter	34
OZReportParameter	35
Parameter	35
Service	36
Servlet	37
SortProperties	43
Versions	44
Viewer	46
2. Interface	50
IReportInfo	50
Part II. OZ Scheduler API	53
1. Class	54
DirectExportResult	54
DirectPrintResult	55
DirectTaskResult	56

FileInfo	57
Program	58
Publisher	60
ScheduledTask	62
Scheduler	64
ServerInfo	74
TaskHolidayGroupInfo	75
TaskHolidayInfo	77
TaskResult	79
2. Interface	82
OZSchedulerUserDefinedPostTaskExecuteEx	82
Part III. OZ Repository API	84
1. Class	85
RepositoryEX	85
Configuring the repository	86
Creating groups and users	87
Editing groups and users	88
Deleting groups and users	90
Getting groups and users	91
Setting up group and user permissions	94
Getting group and user permissions	97
Creating categories and items	100
Editing categories and items	101
Deleting categories and items	105
Getting categories and items	107
Setting up category and item permissions	110
Getting category and item permissions	112
Using item version control	114
2. Interface	121
IOZCategoryInfo	122
IOZCategoryInfoList	123
IOZGroupInfo	124
IOZGroupInfoList	126
IOZHistoryInfo	126
IOZHistoryInfoList	127
IOZItemInfo	128
IOZLoginInfo	130

IOZRepository	132
IOZRepositoryAuthGroupCategory	135
IOZRepositoryAuthGroupItem	138
IOZRepositoryAuthUserCategory	140
IOZRepositoryAuthUserItem	143
IOZRepositoryCategory	145
IOZRepositoryExternalLogin	150
IOZRepositoryGroup	151
IOZRepositoryItem	158
IOZRepositoryItemHistory	164
IOZRepositoryMultiLoginUser	166
IOZRepositoryUser	168
IOZUserInfo	173
IOZUserInfoList	174
 Part IV. User Defined Log API	 176
1. Class	177
OZUserDefinedLogger	177
2. Interface	179
OZClearQuestLogTarget	179
OZDMLogTarget	180
OZEndTarget	181
OZExecuteTarget	182
OZFileStoreLogTarget	183
OZHTTPStoreLogTarget	183
OZMonitorLogTarget	184
OZProcedureLogTarget	186
OZRDBLogTarget	187
OZRequestTarget	188
OZSAPLogTarget	189
OZSchedulerUDLTarget	190
OZSchedulerUserDefinedLogger	191
OZSOAPLogTarget	192
OZTaskInfoTarget	193
OZUDSLogTarget	194
OZUserDefinedLogTarget	195
OZXMLLogTarget	196
 Part V. User Defined Resource API	 197

1. Class	198
OZUDRInfo	198
OZUDRResult	200
Resource	201
2. Interface	204
IOZUserDefinedResource	204
Part VI. User Component API	205

OZ Java Server API

OZ Java Server API provides [classes](#) and [interfaces](#).

Class

OZ Java Server API provides the following classes:

- [Cache](#)
- [ConnectionPool](#)
- [ConnectionPoolInfo](#)
- [ConnectionPoolStatus](#)
- [DataBind](#)
- [HCDataModule](#)
- [HCDataSet](#)
- [Log](#)
- [Mail](#)
- [MaxRowsOfSet](#)
- [MemoryStatus](#)
- [Module](#)
- [Monitor](#)
- [OZCPException](#)
- [OZDataSetInfo](#)
- [OZODIItem](#)
- [OZParameter](#)
- [OZReportParameter](#)
- [Parameter](#)
- [Service](#)
- [Servlet](#)
- [SortProperties](#)

- [Versions](#)
- [Viewer](#)

Cache

This class provides methods related to the cache function of OZ Server.

Constructor Detail

Cache

```
public Cache(String url, String id, String pw, boolean bAutoLogin, boolean useUSL)
```

Performs processing related to OZ Server Cache Manager and processes functions such as the Cache Manager Configuration and Delete Cache Item.

Parameters:

url - URL of OZ Server

id - User ID

pw - User Password

bAutoLogin - Whether to use the autologin

useUSL - Whether to use the USL

Method Detail

getCacheConfiguration

```
public SortProperties getCacheConfiguration() throws OZCPException
```

Gets the cache settings of the OZ Server.(refer to [SortProperties](#))

setCacheConfiguration

```
public void setCacheConfiguration(SortProperties p) throws OZCPException
```

Changes the cache configurations.

Parameters:

p - Cache configuration property(refer to [SortProperties](#))

ConnectionPool

This class provides methods related to the database connection pool of OZ Server.

Constructor Detail

ConnectionPool

```
public ConnectionPool(String url, String id, String pw, boolean bAutoLogin,  
boolean useUSL)
```

An interface for the database connection pool of the OZ Server, performing management functions such as registering, deleting the JDBC/ODBC data of the database.

Parameters:

url - URL of OZ Server

id - User ID

pw - User Password

bAutoLogin - Whether to use the Autologin

useUSL - Whether to use the USL

Method Detail

addPool

```
public void addPool(ConnectionPoolInfo pool) throws OZCPException
```

Adds ConnectionPool. Set a value in the declared member variable in the "ConnectionPoolInfo" class to add ConnectionPool.

Parameters:

pool - ConnctionPoollInfo object, which is the information of the ConnectionPool to add

removePool

```
public void removePool(String pool) throws OZCPException
```

Deletes the ConnectionPool of the selected name.

Parameters:

pool - ConnectionPool Name

getPoolInfoList

```
public ConnectionPoolInfo[] getPoolInfoList() throws OZCPException
```

Gets the ConnectionPoolInfo list for all ConnectionPools.

getPoolStatusList

```
public ConnectionPoolStatus[] getPoolStatusList() throws OZCPException
```

Gets the status information of all ConnectionPools.

getPoolInfo

```
public ConnectionPoolInfo getPoolInfo(String alias) throws OZCPException
```

Gets the ConnectionPoolInfo list for the corresponding ConnectionPool.

Parameters:

alias - Alias of the ConnectionPool

save

```
public void save() throws OZCPException
```

Save the data of all ConnectionPool in the server.

ConnectionPoollInfo

This class provides methods to get the information about database connection.

Method Detail

getAlias

```
public String getAlias()
```

Gets the alias of the connection.

setAlias

```
public void setAlias(String _alias)
```

Sets the alias of the connection.

Parameters:

_alias - Alias

getVendor

```
public String getVendor()
```

Gets the vendor name of the connection.

setVendor

```
public void setVendor(String vendorName)
```

Sets the vendor name of the connection.

Parameters:

vendorName - DB Vendor name

getDriver

```
public String getDriver()
```

Gets the Driver name of the connection.

setDriver

```
public void setDriver(String _driver)
```

Sets the Driver name of the connection.

Parameters:

_driver - Driver name

getURL

```
public String getURL()
```

Gets the URL of the connection.

setURL

```
public void setURL(String _url)
```

Sets the URL of the connection.

Parameters:

_url - URL

getInitConns

```
public int getInitConns()
```

Gets the initial number of connections.

setInitConns

```
public void setInitConns(int _initConns)
```

Sets the initial number of connections.

Parameters:

_initConns - Initial connections

getMaxConns

```
public int getMaxConns()
```

Gets the maximum number of connections.

setMaxConns

```
public void setMaxConns(int _maxConns)
```

Sets the maximum number of connections.

Parameters:

_maxConns - Maximum connections

getTimeout

```
public int getTimeout()
```

Gets the connection timeout in sec.

setTimeout

```
public void setTimeout(int _timeout)
```

Sets the connection timeout.

Parameters:

_timeout - connection timeout

getDoConnectionCheck

```
public boolean getDoConnectionCheck()
```

Gets status of connection validity check

setDoConnectionCheck

```
public void setDoConnectionCheck(boolean _check)
```

Sets status of connection validity check.

Parameters:

_check - Check status

getTestQueryString

```
public String getTestQueryString()
```

Gets the query to test connection.

setTestQueryString

```
public void setTestQueryString(String _testQuery)
```

Sets the query to test connection.

Parameters:

_testQuery - Query to test connection

getInitSqls

```
public String getInitSqls()
```

Gets the initial query to execute after connection open.

setInitSqls

```
public void setInitSqls(String _initQuery)
```

Sets the initial query to execute after connection open.

Parameters:

_initQuery - Initial query

getCloseSqls

```
public String getCloseSqls()
```

Gets the query to execute before closing or freeing a connection.

setCloseSqls

```
public void setCloseSqls(String _closeQuery)
```

Sets the query to execute before closing or freeing a connection.

Parameters:

_closeQuery - Closing query

getSessionQuery

```
public String getSessionQuery()
```

Gets the session query of a connection.

setSessionQuery

```
public void setSessionQuery(String _sessionQuery)
```

Sets the session query of a connection.

Parameters:

_sessionQuery - Session query

getAutoCommit

```
public String getAutoCommit()
```

Gets AutoCommit status.

setAutoCommit

```
public void setAutoCommit(String _commit)
```

Sets AutoCommit status.

Parameters:

_commit - AutoCommit status: "true" or "false"

getProperties

```
public Properties getProperties()
```

Gets the property value.

setProperties

```
public void setProperties(Properties _props)
```

Sets the property value.

Parameters:

_props - properties

getKey

```
public String getKey()
```

Gets the unique Key of the connection. Default key value is Combination of Items Key of the vendor in dbconfig.xml file

setKey

```
public void setKey(String _key)
```

Sets the unique Key of the connection. Setting user defined key value other than default key value is not recommended.

Parameters:

_key - Key value

getItems

```
public HashMap getItems()
```

Gets Item <key,value> of the vendor in dbconfig.xml.

setItems

```
public void setItems(String _map)
```

Sets Item <key,value> of the vendor in dbconfig.xml.

Parameters:

_map - Key value

getDecodeCharSet

```
public void getDecodeCharSet()
```

Gets the decoding set for the corresponding ConnectionPool.

setDecodeCharSet

```
public void setDecodeCharSet(String decode)
```

Sets the decoding set for the corresponding ConnectionPool.

Parameters:

decode - Decoding set for the ConnectionPool

getEncodeCharSet

```
public String getEncodeCharSet()
```

Gets the encoding set for the corresponding ConenctionPool.

setEncodeCharSet

```
public void setEncodeCharSet(String encode)
```

Sets the encoding set for the corresponding ConnectionPool.

Parameters:

encode - Encoding set for the ConnectionPool

getFetchRow

```
public int getFetchRow()
```

Gets the number of JDBC result set rows to get by one time.

setFetchRow

```
public void setFetchRow(int rows)
```

Sets the number of JDBC result set rows to get by one time.

Parameters:

rows - Row count

ConnectionPoolStatus

This class provides member variables and methods to get the state information of the connection pool.

Member Variables

OK

```
public final static int OK
```

ConnectionPool creation was successful.

DRIVER_ERROR

```
public final static int DRIVER_ERROR
```

Failed to load JDBC driver required to create ConnectionPool.

CONNECTION_ERROR

```
public final static int CONNECTION_ERROR
```

Failed to connect DBMS.

Method Detail

getStatusString

```
public String getStatusString()
```

Gets the current status message.

getStatus

```
public int getStatus()
```

Gets the current status.

getFreeConnectionCount

```
public int getFreeConnectionCount()
```

Gets the number of free connections.

getCheckedOutConnectionCount

```
public int getCheckedOutConnectionCount()
```

Gets the number of Checked out connections.

getErrorMessage

```
public String getErrorMessage()
```

Gets the error message occurred during ConnectionPool creation.

DataBind

This class provides methods related to the data binding function of OZ Server.

Constructor Detail

DataBind

```
public DataBind(String url, String id, String pw, boolean bAutoLogin, boolean  
useUSL)
```

Performs data binding function

Parameters:

url - URL of OZ Server

id - User ID

pw - User Password

bAutoLogin - Whether to use the AutoLogin

useUSL - Whether to use the USL

Method Detail

getBindConfiguration

```
public SortProperties getBindConfiguration() throws OZCPException
```

Gets the values defined in the databind.properties file.

setBindConfiguration

```
public void setBindConfiguration(SortProperties config) throws OZCPException
```

Sets the values defined in the databind.properties file.

Parameters:

config - DataBind configuration properties

HCDataModule

This class provides member variable and methods to get the information of the HCDataModule.

Member Variables

PARAMETERSET_NAME

```
public final static String PARAMETERSET_NAME
```

Type of OZ parameter.

Method Detail

getDataSetNames

```
public String[] getDataSetNames()
```

Gets the names of all data sets.

getDataSets

```
public HCDataSet[] getDataSets(String name)
```

Gets the data set with the corresponding name. The data set name makes a distinction between capitals and small letters. If you set the detail set name in the mater-detail structure, the master set is also returned. If there is no data set with the name, returns an array of which the length is 0.

Parameters:

name - Name of Data set

PrintInfo

```
public void PrintInfo(PrintStream out)
```

Prints the data and field information as PrintStream.

Parameters:

out - PrintStream object to print

HCDataSet

This class provides method to get the information of the HCDataSet.

Method Detail

getResultSet

```
public ResultSet getResultSet()
```

Gets the result set.

Log

This class provides methods related to the log function of OZ Server.

Constructor Detail

Log

```
public Log(String url, String id, String pw, boolean bAutoLogin, boolean useUSL)
```

Performs the processing related to OZ server Log Manager and configures the settings for recording the various messages, information, and errors, etc. arising during server operation.

Parameters:

url - URL of OZ Server

id - User ID

pw - User Password

bAutoLogin - Whether to use the AutoLogin

useUSL - Whether to use the USL

Method Detail

getConfigure

```
public String getConfigure() throws OZCPException
```

Gets the log configurations from the server.

downloadLog

```
public byte[] downloadLog() throws OZCPException
```

Downloads the log file from the server.

downloadLog

```
public void downloadLog(String fileName) throws OZCPException
```

Saves the log file downloaded from the server to the local.

Parameters:

fileName - Name of the file to save

setConfigure

```
public void setConfigure(String logs) throws OZCPException
```

Changes the log configuration values.

Parameters:

logs - The log configuration value to modify, set in the form of "key=value"

setPriority

```
public void setPriority(String p) throws OZCPException
```

Sets the level of the log.(INFO, DEBUG, ERROR)

Parameters

p - The log level to change

Mail

This class provides methods related to the email sending function of OZ Server.

Constructor Detail

Mail

```
public Mail(String url, String id, String pw, boolean bAutoLogin, boolean useUSL)
```

Performs email sending function

Parameters:

url - URL of OZ Server

id - User ID

pw - User Password

bAutoLogin - Whether to use the AutoLogin

useUSL - Whether to use the USL

Method Detail

addAlias

```
public void addAlias(String aliasName, SortProperties p) throws OZCPEException
```

Adds mail configuration information with the alias configured in the OZ Server.

Parameters:

aliasName - The alias

p - The email configuration property value(refer to [SortProperties](#))

getAliasConfig

```
public SortProperties getAliasConfig(String aliasName) throws OZCPException
```

Gets the mail configuration information of the corresponding alias.

Parameters:

aliasName - The alias

getMailAliasNames

```
public String[] getMailAliasNames()
```

Gets all mail aliases configured in the amil.properties in the OZ Server.

modifyAlias

```
public void modifyAlias(String aliasName, String newAliasName, SortProperties p)  
throws OZCPException
```

Modifies the name and mail configuration information of the corresponding alias.

Parameters:

aliasName - The alias

newAliasName - The new alias

p - Mail configuration property value

removeAlias

```
public void removeAlias(String aliasName) throws OZCPException
```

Deletes the mail configuration information of the corresponding alias.

Parameters:

aliasName - The alias

send

```
public void send(String aliasName, String from, String fromUserName, String to,  
String cc, String bcc, String subject, String context, boolean isHTML, String  
localFilePath, String fileName) throws OZCPEException
```

Sends asynchronous mail according to the mail information of the corresponding alias.

Parameters:

aliasName - The alias

from - The email address of a person who sent the email

fromUserName - The name who sent the email

to - The email address of a person to receive the email

cc - The cc email address

bcc - The bcc email address

subject - The email subject

context - The email content

isHTML - Whether to set the message format to HTML

localFilePath - The full path of the file for attachment

fileName - The attached file name

sendSync

```
public boolean sendSync(String aliasName, String from, String fromUserName, String  
to, String cc, String bcc, String subject, String context, boolean isHTML, String  
localFilePath, String fileName) throws OZCPEException
```

Sends synchronous mail according to the mail information of the corresponding alias, and gets whether the transmission succeeded.

Parameters:

aliasName - The alias

from - The email address of a person who sent the email

fromUserName - The name who sent the email

to - The email address of a person to receive the email

cc - The cc email address

bcc - The bcc email address

subject - The email subject

context - The email content

isHTML - Whether to set the message format to HTML

localFileFullPath - The full path of the file for attachment

fileName - The attached file name

Mail > SortProperties

The keys supported by SortProperties are as follows:

Key List

Key	Description
active	Whether to use mail sending function
fromSend	The email address of a person who sent the email
toSend	The email address of a person to receive the email
SMTPServer	SMTP Server URL
SMTPServerPort	SMTP Server Port Number

Key	Description
SMTPUserID	User ID to be authenticated when using a mail server that requires SMTP authentication
SMTPUserPassword	When using a mail server that requires SMTP authentication, the password of the user ID to be authenticated
SMTPUserID_encrypted	Encrypted user ID to authenticate when using a mail server that requires SMTP authentication
SMTPUserPassword_encrypted	Encrypted user password to authenticate when using a mail server that requires SMTP authentication
EnableSSL	Whether to use SSL
SendRetryCount	Number of retransmissions in case of mail transmission failure
SendRetryPeriodTime	Resending cycle in case of mail transmission failure(Unit: Seconds)
PrefixSubjectMessage	Prefix to use before the mail subject

MaxRowsOfSet

This class provides methods related to the MaxRows property of the data set.

Method Detail

getSetName

```
public String getSetName()
```

Gets the name of data set that the Maximum Rows property value is set.

setSetName

```
public void setSetName(String v)
```

Sets the name of data set to set the Maximum Rows property value.

Parameters:

v - Name of Data set

getMaxRow

```
public int getMaxRow()
```

Gets value of Maximum Rows property.

setMaxRow

```
public void setMaxRow(int v)
```

Sets value of Maximum Rows property.

Parameters:

v - Value of Maximum Rows property

MemoryStatus

This class provides methods to get the information about the system's memory.

Method Detail

garbageCollect

```
public static void garbageCollect()
```

Do garbage collection.

getTotalMemory

```
public long getTotalMemory()
```

Gets total memory of the server VM.

getFreememory

```
public long getFreememory()
```

Gets free memory of the server VM.

Module

This class provides methods related to the data module processing function of OZ Server.

Constructor Detail

Module

```
public Module(String url, String id, String pw, boolean bAutoLogin, boolean  
useUSL)
```

Performs OZ Data Module related functions.

Parameters:

url - URL of OZ Server

id - User ID

pw - User Password

bAutoLogin - Whether to use the AutoLogin

useUSL - Whether to use the USL

Method Detail

addODIParameter

```
public void addODIParameter(String odiName, String key, String value)
```

Configures the ODI parameter values to create SDM files. Default ODI parameter values will be used if not configured.

Parameters:

odiName - ODI name

key - ODI parameter name

value - ODI parameter value

addODIParameter

```
public void addODIParameter(String odiName, String item, String category,  
Hashtable paramHash) throws IllegalArgumentException
```

Configures the ODI parameter values to create SDM files. Default ODI parameter values will be used if not configured. Used when creating SDM generation units for each parameter and when each different SDM file is needed depending on the ODI parameter values.

Parameters:

odiName - ODI name set in the Designer

item - ODI file name

category - ODI file path

paramHash - Hashtable where the parameter keys and values are set.

addParameter

```
public void addParameter(String key, String value)
```

Configures form parameter values to create SDM files. Default form parameter values will be used if not configured.

Parameters:

key - Form Parameter Name

value - Form Parameter Value

addReport

```
public IReportInfo addReport(String itemName, String categoryName) throws  
OZCPException
```

Adds reports in the process of making multiple reports into one OZD file.

Parameters:

itemName - Item name where reports will be added

categoryName - Category name of the item where reports will be added

addReport

```
public IReportInfo addReport(String itemName, String categoryName, String  
displayName) throws OZCPException
```

Adds reports in the process of making multiple reports into one OZD file.

Parameters:

itemName - Item name where reports will be added

categoryName - Category name of the item where reports will be added

displayName - Report name to appear in the report tree of the OZ Viewer

getODIParameter

```
public OZParameter[] getODIParameter(String category, String odiName) throws  
OZCPException
```

Gets ODI parameter from ODI

Parameters:

category - Category name

odiName - ODI name

getOZD

```
public InputStream getOZD() throws OZCPException
```

Create an OZD with multiple reports and gets its Input Stream.

getOZD

```
public InputStream getOZD(String item, String category, String[] urls) throws  
OZCPException
```

The first type function is a function that gets a report file from the server to create an OZD file.

Parameters:

item - Item name(OZR file name)

category - Category name

urls - The URL of the image file to save together with the OZD

getOZQReportParameter

```
public OZODIItem[] getOZQReportParameter(String category, String ozqItemName)  
throws OZCPException
```

Gets ODI parameter from OZ Query Designer.

Parameters:

category - Category name

ozqItemName - ODI name

getOZReportParameter

```
public OZReportParameter getOZReportParameter(String category, String reportName)  
throws OZCPEException
```

Gets ODI parameter and form parameter from report.

Parameters:

category - Category name

reportName - Report name

getOZReportDataSetInfo

```
public OZDataSetInfo[] getOZReportDataSetInfo(String category, String reportName)  
throws OZCPEException
```

Gets the dataset information in the report. Returns in the order they are added to the data tree.

Parameters:

category - Category name

reportName - Report name

registODIPath

```
public void registODIPath(String odiName, String path) throws  
IllegalArgumentException
```

Registers an ODI file to dynamically change when creating an OZD.

Parameters:

odiName - ODI name

path - ODI file path

saveOZD

```
public void saveOZD(String fileName) throws OZCPEException
```

Creates an OZD with multiple reports and saves it in a specific path.

Parameters:

fileName - Category path where the file will be saved and file name

saveOZD

```
public void saveOZD(String fileName, String item, String category, String[] urls)  
throws OZCPEException
```

Saves a report file as OZD file.

Parameters:

fileName - OZD file name including the save path

item - Report file(.ozr) name

category - Category name of the report file(.ozr)

urls - URL of the image file to save together with the OZD file

setMemoAllowed

```
public void setMemoAllowed(boolean allowed)
```

Sets whether to allow memo or not.

Parameters:

allowed - True or false

setPassword

```
public void setPassword(String password)
```

Set password.

Parameters:

password - Password

Monitor

This class provides methods related to the monitor log function of OZ Server.

Constructor Detail

Monitor

```
public Monitor(String url, String id, String pw, boolean bAutoLogin, boolean  
useUSL)
```

Performs the real-time monitoring related functions of OZ Server.

Parameters:

url - URL of OZ Server

id - User ID

pw - User Password

bAutoLogin - Whether to use the AutoLogin

useUSL - Whether to use the USL

Method Detail

getVersions

```
public Versions getVersions() throws OZCPEException
```

Gets the version information of the server and respective modules.

getMemoryInfo

```
public MemoryStatus getMemoryInfo() throws OZCPEException
```

Gets the current memory usage status.(total memory, used memory, free memory)

downloadMonitorLog

```
public byte[] downloadMonitorLog() throws OZCPEException
```

Gets the monitor log file saved in the Server.

downloadMonitorLog

```
public void downloadMonitorLog(String fileName) throws OZCPException
```

Gets the monitor log file saved in the Server and saves it in the set location as file name.

OZCPException

This class provides methods related to the exception that occurs while using API.

Method Detail

getErrorCode

```
public int getErrorCode()
```

Gets the error code.

getMessage

```
public String getMessage()
```

Gets the error message.

OZDataSetInfo

This class provides methods to get the information about data set added to the OZR file.

Method Detail

getDataSetServiceType

```
public String getDataSetServiceType()
```

Gets the service type. Returns one of ODI, DataService, FXData.

getDataSetName

```
public String getDataSetName()
```

Gets the dataset name.

getDataSetFields

```
public OZAttributeList getDataSetFields()
```

Gets the dataset field information.

getMaxRowCount

```
public int getMaxRowCount()
```

Gets the value of "Max Row Count" property.

getUse

```
public boolean getUse()
```

Gets the value of "Use" property.

getExportable

```
public boolean getExportable()
```

Gets the value of "Exportable" property.

OZODIItem

This class provides methods to get the information about ODI file.

Method Detail

getCategory

```
public String getCategory()
```

Gets the category name.

getODIName

```
public String getODIName()
```

Gets the ODI name.

getODIFileName

```
public String getODIFileName()
```

Gets the ODI file name.

getODIParameters

```
public OZParameter[] getODIParameters()
```

Gets the ODI parameter.

OZParameter

This class provides methods to get the information about FORM parameters added to the OZR file.

Method Detail

getName

```
public String getName()
```

Gets the parameter name.

getType

```
public int getType()
```

Gets the parameter type.

getValue

```
public String getValue()
```

Gets the parameter value.

getDescription

```
public String getDescription()
```

Gets the parameter description.

OZReportParameter

This class provides methods to get the information about ODI parameters and FORM parameters added to the OZR file.

Method Detail

getODIList

```
public OZODIItem[] getODIList()
```

Gets the ODI information object.

getFormParameters

```
public OZParameter[] getFormParameters()
```

Gets the form parameter information object.

Parameter

This class provides member variables to get the information about ODI parameters.

Member Variables

name

```
public String name
```

Parameter name.

value

```
public String value
```

Parameter value.

Service

This class provides methods related to the service function of OZ Server.

Constructor Detail

Service

```
public Service(String url, String id, String pw, boolean bAutoLogin, boolean  
useUSL)
```

Performs processing functions such as OZ Server information, status, processing, memory collection.

Parameters:

url - URL of OZ Server

id - User ID

pw - User password

bAutoLogin - Whether to use the AutoLogin

useUSL - Whether to use the USL

Method Detail

garbageCollect

```
public void garbageCollect() throws OZCPEException
```

Runs garbage collection, which collects the free memory on the JVM using the separate memory collection program so that the memory can be recycled.

ping

```
public boolean ping() throws OZCPEException
```

Returns whether the server is operating.

getHandlerCount

```
public int getHandlerCount() throws OZCPEException
```

Gets the number of concurrent multithread of the server.

Servlet

This class provides methods related to the OZD file generation function of OZ Server.

Method Detail

getOZD

```
public final InputStream getOZD(String item, String category, Hashtable formparam,  
Hashtable odiparam, boolean memoallowed, String password, String id, String pwd,  
HttpServletRequest request) throws Exception
```

Inherits an OZ Servlet server and generates one that creates OZDs. Communicates the newly generated OZ Servlet server to create OZD files.

Parameters:

item - Item name(OZR file name)

category - Category name

formparam - Form parameter

odiparam - ODI parameter

memoallowed - Password to apply when opening the OZD document

password - User ID to log in to the OZ Server

id - User password to log in to the OZ Server

pwd - HttpServletRequest

request - Report name

getOZD

```
public final InputStream getOZD(String item, String category, String  
serverDMType, Hashtable formparam, Hashtable odiparam, Hashtable odipath, boolean  
memoallowed, String password, String id, String pwd, HttpServletRequest request)
```

Inherits an OZ Servlet server and generates one that creates OZDs. Communicates the newly generated OZ Servlet server to create OZD files.

Parameters:

item - Item name(OZR file name)

category - Category name

serverDMType - Set whether to process the data module received from the OZ Server as memory or file.(default: memory)

formparam - Form parameter

odiparam - ODI parameter

odipath - Set whether to allow making notes in the OZD document

memoallowed - Password to apply when opening the OZD document

password - User ID to log in to the OZ Server

id - User password to log in to the OZ Server

pwd - HttpServletRequest

request - Report name

getOZD

```
public final InputStream getOZD(String[] item, String[] category, Hashtable[]  
formparam, Hashtable[] odiparam, String id, String pwd, HttpServletRequest  
request, String[] displayName, String[][] urls, Hashtable[] options) throws  
Exception
```

Inherits an OZ Servlet server and generates one that creates OZDs. Communicates the newly generated OZ Servlet server to create OZD files.

Parameters:

item - Item name(OZR file name)

category - Category name

formparam - Form parameter

odiparam - ODI parameter

id - User password to log in to the OZ Server

pwd - HttpServletRequest

request - Report name

displayName - Image file URL

urls - OZD save options

options - Set whether to allow making notes in the OZD document

getOZD

```
public final InputStream getOZD(String[] item, String[] category, Hashtable[]  
formparam, Hashtable[] odiparam, boolean memoallowed, String password, String id,  
String pwd, HttpServletRequest request, String[] displayName, String[][] urls,  
Hashtable[] options) throws Exception
```

Inherits an OZ Servlet server and generates one that creates OZDs. Communicates the newly generated OZ Servlet server to create OZD files.

Parameters:

item - Item name(OZR file name)

category - Category name

formparam - Form parameter

odiparam - ODI parameter

memoallowed - Password to apply when opening the OZD document

password - User ID to log in to the OZ Server

id - User password to log in to the OZ Server

pwd - HttpServletRequest

request - Report name

displayName - Image file URL

urls - OZD save options

options - Set whether to allow making notes in the OZD document

getOZD

```
public final InputStream getOZD(String[] item, String[] category, String  
serverDMType, Hashtable[] formparam, Hashtable[] odiparam, boolean memoallowed,  
String password, String id, String pwd, HttpServletRequest request, String[]  
displayName, String[][] urls, Hashtable[] options) throws Exception
```

Inherits an OZ Servlet server and generates one that creates OZDs. Communicates the newly generated OZ Servlet server to create OZD files.

Parameters:

item - Item name(OZR file name)

category - Category name

serverDMType - Set whether to process the data module received from the OZ Server as memory or file.(default: memory)

formparam - Form parameter

odiparam - ODI parameter

memoallowed - Password to apply when opening the OZD document

password - User ID to log in to the OZ Server

id - User password to log in to the OZ Server

pwd - HttpServletRequest

request - Report name

displayName - Image file URL

urls - OZD save options

options - Set whether to allow making notes in the OZD document

getOZD

```
public final InputStream getOZD(String[] item, String[] category, String serverDMType, Hashtable[] formparam, Hashtable[] odiparam, Hashtable[] odiPaths, boolean memoallowed, String password, String id, String pwd, HttpServletRequest request, String[] displayName, String[][] urls, Hashtable[] options, Hashtable[] fxURLs) throws Exception
```

Inherits an OZ Servlet server and generates one that creates OZDs. Communicates the newly generated OZ Servlet server to create OZD files.

Parameters:

item - Item name(OZR file name)

category - Category name

serverDMType - Set whether to process the data module received from the OZ Server as memory or file.(default: memory)

formparam - Form parameter

odiparam - ODI parameter

odiPaths - Set whether to allow making notes in the OZD document

memoallowed - Password to apply when opening the OZD document

password - User ID to log in to the OZ Server

id - User password to log in to the OZ Server

pwd - HttpServletRequest

request - Report name

displayName - Imgae file URL

urls - OZD save options

options - Set whether to allow making notes in the OZD document

fxURLs - Framework URL of FX data and character set

getOZDWithSDM

```
public final InputStream getOZDWithSDM(String item, String category, Hashtable sdm, Hashtable formParam, boolean memoallowed, String password, String id, String pwd, HttpServletRequest request) throws Exception
```

Gets a report file and SDM file and creates and gets an OZD file.

Parameters:

item - Item name(OZR file name)

category - Category name

sdm - Information of the SDM that will create the OZD file

formparam - Form parameter

memoallowed - Whether to allow making notes in the OZD document

password - The password to apply when opening the OZD document

id - User ID to log in to the OZ Server

pwd - User password to log in to the OZ Server

request - HttpServletRequest

SortProperties

This class provides methods related to getCacheConfiguration and setCacheConfiguration.

Method Detail

get

```
public synchronized Object get(String key)
```

Gets the value of the key.

Parameters:

key - Name of key

getProperty

```
public synchronized String getProperty(String key)
```

Gets the value of the key.

Parameters:

key - Name of key

setProperty

```
public synchronized Object setProperty(String key, String value)
```

Sets the value of the key.

Parameters:

key - Name of the key

value - Value of the key

put

```
public synchronized Object put(Object key, Object value)
```

Sets value of the key.

Parameters:

key - Name of the key

value - Value of the key

The keys that can be set are as follows:

key	Description
Active	Whether to use cache key
CACHE_FILE_PATH	Cache storage path
DM_CACHE_FILE_PATH	Data Module cache storage path
memoryCacheValidTime	Memory cache timeout(unit: seconds)
diskCacheValidTime	Disk Cache Timeout (unit: seconds)
FreeMemoryPercentage	Free memory percentage

Versions

This class provides member variables to get the information of OZ Server and the system on which OZ Server is running.

Member Variables

osName

`public String osName`

OS name of the system.

osVersion

`public String osVersion`

OS version of the system.

javaVendor

`public String javaVendor`

Vendor of Java set in WAS connected with OZ Server.

javaVersion

`public String javaVersion`

Version of Java set in WAS connected with OZ Server.

OZServerVersion

`public String OZServerVersion`

Version of OZ Server.

CPRelease

`public String CPRelease`

Release information of OZ Common Protocol.

CPProtocol

`public int CPPProtocol`

Version of OZ Common Protocol.

DMRelease

`public String DMRelease`

Release information of OZ Data Module.

DMStreaming

`public int DMStreaming`

Streaming protocol version of OZ Data Module.

Viewer

This class provides methods related to the OZ Server's ability to process form files and data requested by OZ Viewer.

Constructor Detail

Viewer

`public Viewer(String url, String id, String pw, boolean bAutoLogin, boolean useUSL)`

Performs the report file and data processing requested by OZ Viewer.

Parameters:

url - URL of OZ Server

id - User ID

pw - User Password

bAutoLogin - Whether to use the AutoLogin

useUSL - Whether to use the USL

Method Detail

getByteArrayForm

```
public byte[] getByteArrayForm(String reportName, String categoryName) throws  
OZCPException
```

Gets the content of the report.

Parameters:

reportName - Report name

categoryName - Category name of the report

getForm

```
public InputStream getForm(String reportName, String categoryName, boolean  
isCompress) throws OZCPException
```

Gets the content of the report after setting whether to compress it.

Parameters:

reportName - Report name

categoryName - Category name of the report

isCompress - Whether or not compressed

getUserParametersWithValue

```
public Parameter[] getUserParametersWithValue(String itemName, int  
itemType, String categoryName) throws OZCPException
```

Informs of the parameters used in the item and its default values.

Parameters:

itemName - Item name

itemType - Item type(OZR: 20001 / ODI: 10000)

categoryName - Category name of the item

isCompress - Whether or not compressed

getHCUSDM

```
public HCDataModule getHCUSDM(String fileName, String categoryName) throws  
OZCPException
```

Gets the SDM saved in the server to the data.

Parameters:

fileName - Report name

categoryName - Category name of the report

getDataModule

```
public HCDataModule getDataModule(InputStream sdmInput) throws OZCPException
```

Gets the data record from the SDM.

Parameters:

sdmInput - SDM stream

getDataModule

```
public HCDataModule getDataModule(String odiName, String categoryName, Parameter[]  
parameters, boolean doCompress, boolean forceRefresh) throws OZCPException
```

Gets the selected data record.

Parameters:

odiName - Name of the data source information file

categoryName - Category name of the data source file

parameters - User parameter to get the data record

doCompress - Whether to transmit compressed

forceRefresh - Whether to use the cache in the server

getDataModule

```
public HCDataModule getDataModule(String odiName, String categoryName, Parameter[] parameters, boolean doCompress, boolean forceRefresh, String[] invalidDataset, MaxRowsOfSet[] maxRows) throws OZCPException
```

Gets the selected data record.

Parameters:

odiName - Name of the data source information file

categoryName - Category name of the data source file

parameters - User parameter to get the data record

doCompress - Whether to transmit compressed

forceRefresh - Whether to use the cache in the server

invalidDataset - Name of the unused data set among the data sources

maxRows - Maximum number of records of the data set

Interface

OZ Java Server API provides the following Interface:

- [IReportInfo](#)

IReportInfo

This interface provides a method to set additional information about a report when generating an OZD with multiple reports.

Method Detail

getItemName

```
public String getItemName()
```

Gets the item name.

getCategoryName

```
public String getCategoryName()
```

Gets the category name of the item.

addURL

```
public void addURL(String[] urls)
```

Sets the URL of the image file included in the report as array.

Parameters:

urls - Image file URL

addURL

```
public void addURL(String url)
```

Sets the URL of the image file included in the report.

Parameters:

url - Image file URL

addParameter

```
public void addParameter(String key, String value)
```

Configures the parameter values.

Parameters:

key - Parameter name

value - Parameter value

addODIParameter

```
public void addODIParameter(String odiName, String key, String value)
```

Configures the ODI parameter values.

Parameters:

odiName - ODI name

key - ODI parameter name

value - ODI parameter value

addProperties

```
public void addProperties(String key, String value)
```

Configures report save options.

Parameters:

key - Property name

value - Property value

addFrameworkURL

`public void addFrameworkURL(String fxName, String url, String charset)`

Configures the framework URL of FX data.

Parameters:

fxName - FX data name

url - Framework URL

charset - Character set for parameter of framework URL

OZ Scheduler API

OZ Scheduler API provides [classes](#) and [interfaces](#).

Class

OZ Scheduler API provides the following classes:

- [DirectExportResult](#)
- [DirectPrintResult](#)
- [DirectTaskResult](#)
- [FileInfo](#)
- [Program](#)
- [Publisher](#)
- [ScheduledTask](#)
- [Scheduler](#)
- [ServerInfo](#)
- [TaskHolidayGroupInfo](#)
- [TaskHolidayInfo](#)
- [TaskResult](#)

DirectExportResult

This class provides the execution result of a task whose execution type is DirectExport.

Member Variables

exportFileList

```
public String exportFileList
```

List of exported file names. Multiple file names are listed with " | " as a separator.

pageCount

```
public int pageCount
```

Number of exported pages.

DirectPrintResult

This class provides the execution result of a task whose execution type is DirectPrint.

Member Variables

pageCount

```
public int pageCount
```

Number of printed pages.

pageCopy

```
public int pageCopy
```

Number of copies.

pageRange

```
public String pageRange
```

Range of printed page.

printerName

```
public String printerName
```

Printer name.

printerDriverName

```
public String printerDriverName
```

Printer driver name.

DirectTaskResult

This class provides the execution result of a task whose execution type is DirectExport or DirectPrint.

Member Variables

taskID

```
public String taskID
```

ID of the task.

taskName

```
public String taskName
```

Name of the task.

taskGoupName

```
public String taskGoupName
```

Group name of the task.

completedTime

```
public String completedTime
```

Completion time of task.

executeTime

```
public String executeTime
```

Run time of task.

isSuccessful

```
public boolean isSuccessful
```

Whether the task was successful.

formCategoryName

```
public String formCategoryName
```

Category name of the OZR file.

formName

```
public String formName
```

Name of the OZR file.

errorMessage

```
public String errorMessage
```

Error message.

FileInfo

This class provides information about files and folders.

Member Variables

isDirectory

```
public boolean isDirectory
```

Whether it is a directory.

name

```
public String name
```

Name of the file or directory.

size

```
public long size
```

Size of the file or directory.

lastModified

```
public long lastModified
```

Time of the last modification of the file or directory, calculated as a date and time based on 00:00:00 GMT on January 1, 1970.

Program

This class is related to the external program registration and management functions of OZ Scheduler.

Constructor Detail

Program

```
public Program(String ip, int port)
```

Performs external program registration and management functions in OZ Scheduler.

Parameters:

ip - IP address

port - Port number

Method Detail

createFolder

```
public void createFolder(ServerInfo s, String folder)
```

Creates an external program folder in the OZ Scheduler.

Parameters:

s - OZ Server information

folder - Name of the folder to create

downloadFile

```
public byte[] downloadFile(ServerInfo s, String file)
```

Downloads the registered external program files.

Parameters:

s - OZ Server information

file - Name of the file to download

getExternalProgramList

```
public FileInfo[] getExternalProgramList(ServerInfo s, String folder)
```

Gets the list of external programs in the OZ Scheduler.

Parameters:

s - OZ Server information

folder - Folder name

removeFiles

```
public void removeFiles(ServerInfo s, String folder, String[] files)
```

Deletes a registered external program file.

Parameters:

s - OZ Server information

folder - Folder name

files - Name of the program file to remove

removeFolder

```
public void removeFolder(ServerInfo s, String folder, boolean isAll)
```

Deletes an external program folder in the OZ Scheduler.

Parameters:

s - OZ Server information

folder - Name of the folder to delete

isAll - Whether to execute methods

uploadFile

```
public void uploadFile(ServerInfo s, String file, byte[] b)
```

Register(upload) an external program in the OZ Scheduler.

Parameters:

s - OZ Server information

file - Name of the file to register

b - Byte of the file to register

Publisher

This is a class that manages the result files of tasks generated by the OZ Scheduler.

Constructor Detail

Publisher

```
public Publisher(String ip, int port)
```

Manage the results file of tasks generated by the OZ Scheduler.

Parameters:

ip - IP address

port - Port number

Method Detail

createFolder

```
public void createFolder(ServerInfo s, String folder)
```

Creates a folder for saving the OZ Scheduler export file.

Parameters:

s - OZ Server information

folder - Name of the created folder

downloadFile

```
public byte[] downloadFile(ServerInfo s, String file)
```

Downloads the registered the OZ Scheduler export file.

Parameters:

s - OZ Server information

file - Downloaded file name

getPublishedFiles

```
public FileInfo[] getPublishedFiles(ServerInfo s, String folder)
```

Gets the list of export files from the OZ Scheduler.

Parameters:

s - OZ Server information

folder - Folder name

removeFiles

```
public void removeFiles(ServerInfo s, String folder, String[] files)
```

Deletes the registered the OZ Scheduler export file.

Parameters:

s - OZ Server information

folder - Deleted folder name

files - Names of deleted files

removeFolder

```
public void removeFolder(ServerInfo s, String folder, boolean isAll)
```

Deletes the folder where the OZ Scheduler export file is saved.

Parameters:

s - OZ Server information

folder - Deleted folder name

isAll - Whether to execute methods

ScheduledTask

This class provides task information.

Member Variables

taskID

```
public String taskID
```

ID of the task.

taskName

```
public String taskName
```

Name of the task.

taskGoupName

```
public String taskGoupName
```

Group name of the task.

formCategoryName

```
public String formCategoryName
```

Category name of the OZR file.

formFileName

```
public String formFileName
```

Name of the OZR file.

schedulingType

```
public int schedulingType
```

Schedule type code.(0: Immediately, 1: Once at specific time, 2: Periodically)

schedulingTypeStr

```
public String schedulingTypeStr
```

Schedule type.(Immediately, Once at specific time, Periodically)

nextRunTimeStr

```
public String nextRunTimeStr
```

Next execution time of task.

lastRunTimeStr

```
public String lastRunTimeStr
```

Last execution time of task.

status

```
public String status
```

Status of task.(W: Wait, R: Run, P: Pause)

Scheduler

This class creates report scheduling jobs and performs the result file and scheduler management functions.

Constructor Detail

Scheduler

```
public Scheduler(String ip, int port)
```

Creates report schedule jobs and perform result file and scheduler management functions.

Parameters:

ip - IP address

port - Port number

Method Detail

createTask

```
public String createTask(ServerInfo s, SortProperties configMap, SortProperties exportMap)
```

Creates a scheduling task based on the form file and gets the task ID.

Parameters:

s - OZ Server information

configMap - Configuration option used when creating task

exportMap - Configuration option of the files exported from the OZ Scheduler

getTask

```
public Vector getTask(ServerInfo s)
```

Gets the result of the currently Scheduling task.

Parameters:

s - OZ sever information

getTaskResult

```
public TaskResult[] getTaskResult(ServerInfo s, String from, String to, String taskId, int rollingIndex)
```

Gets the results of task execution of a rolled file.

Parameters:

s - OZ Server information

from - Starting time

to - End time

taskId - Task ID

rollingIndex - File index

getTaskResults

```
public TaskResult[] getTaskResults(ServerInfo s, String from, String to, String taskId)
```

Gets the scheduling task result.

Parameters:

s - OZ Server information

from - Starting time

to - End time

taskId - Task ID

getTaskProperties

```
public SortProperties[] getTaskProperties(ServerInfo s, String taskId)
```

Gets task properties.

Parameters:

s - OZ Server information

taskId - Task ID

getTaskWaitTime

```
public int getTaskWaitTime(ServerInfo s, String taskId)
```

Gets the amount of time in seconds the task waited.

Parameters:

s - OZ Server information

taskId - Task ID

getTaskWaitCount

```
public int getTaskWaitCount()
```

Gets the number of tasks that are waiting.

removeTask

```
public void removeTask(ServerInfo s, String taskId)
```

Deletes a scheduling task.

Parameters:

s - OZ Server information

taskId - Task ID

modifyTask

```
public String modifyTask(ServerInfo s, String taskId, SortProperties configMap,  
SortProperties exportMap)
```

Modifies a task property.

Parameters:

s - OZ Server information

taskId - Task ID

configMap - OZ Scheduler configuration option

exportMap - Export information

taskPause

```
public boolean taskPause(ServerInfo s, String taskId)
```

Pauses a scheduling task.

Parameters:

s - OZ Server information

taskId - Task ID

taskResume

```
public boolean taskResume(ServerInfo s, String taskId)
```

Resumes a paused scheduling task.

Parameters:

s - OZ Server information

taskId - Task ID

stop

```
public void stop(ServerInfo s, boolean waitTaskId)
```

Stops a scheduler.

Parameters:

s - OZ Server information

waitTaskId - Whether to stops the OZ Scheduler abort

export

```
public boolean export(ServerInfo s, SortProperties configMap, SortProperties  
exportMap)
```

Export by using the viewer parameter as it is and then returns the result.

Parameters:

s - OZ Server information

configMap - OZ Scheduler configuration option

exportMap - Export information

makePDF

```
public boolean makePDF(ServerInfo s, SortProperties configMap, SortProperties  
exportMap)
```

Performs PDF exporting without registering in the OZ Scheduler's work list and then returns the result.

Parameters:

s - OZ Server information

configMap - Form information for export

exportMap - PDF export information

print

```
public boolean print(ServerInfo s, SortProperties configMap, SortProperties  
printMap)
```

Prints by applying the viewer parameter.

Parameters:

s - OZ Server information

configMap - OZ Scheduler configuration option

printMap - Print information

getCurrentStatus

```
public String getCurrentStatus() throws SchedulerException
```

Gets the scheduler's used memory, total memory, number of running viewers, and number of waiting viewers.

getConfiguration

```
public SortProperties getConfiguration(ServerInfo s)
```

Gets the OZ Scheduler configurations.

Parameters:

s - OZ Server information

modifyConfiguration

```
public void modifyConfiguration(ServerInfo s, SortProperties configMap)
```

Modifies the OZ Scheduler configurations.

Parameters:

s - OZ Server information

configMap - OZ Scheduler configuration option

setReceiveTimeout

```
public void setReceiveTimeout(int timeout)
```

Sets the socket timeout.

Parameters:

timeout - Socket timeout

ping

```
public boolean ping()
```

Returns whether the server is running.

getOZSList

```
public String[] getOZSList(String path)
```

Gets the list of OZS files in the set path.

Parameters:

path - The path

ozsVerConv

```
public void ozsVerConv(String oldPath, String newPath)
```

Converts a 2.5 version OZS file into the latest version and saves it.

Parameters:

oldPath - The OZS file name or path to convert

newPath - The path for saving the converted OZS file

addTaskHolidayInfo

```
public boolean addTaskHolidayInfo(TaskHolidayInfo value)
```

Adds task holiday information.

Parameters

value - Task holiday information

modifyTaskHolidayInfo

```
public boolean modifyTaskHolidayInfo(String old_key, TaskHolidayInfo new_value)
```

Modifies task holiday information.

Parameters:

old_key - The task holiday name to modify

new_value - New task holiday information

deleteTaskHolidayInfo

```
public boolean deleteTaskHolidayInfo(String[] keys)
```

Deletes task holiday information.

Parameters:

keys - The task holiday name array to delete

addTaskHolidayGroupInfo

```
public boolean addTaskHolidayGroupInfo(TaskHolidayGroupInfo value)
```

Adds task holiday group information.

Parameters:

value - Task holiday group information

modifyTaskHolidayGroupInfo

```
public boolean modifyTaskHolidayGroupInfo(String old_key, TaskHolidayGroupInfo new_value)
```

Modifies task holiday group information.

Parameters:

old_key - The task holiday group name to modify

new_value - New task holiday group information

deleteTaskHolidayGroup

```
public boolean deleteTaskHolidayGroupInfo(String key)
```

Deletes task holiday group information.

Parameters:

key - The task holiday group name to delete

getTaskHolidayInfoList

```
public OZMap getTaskHolidayInfoList()
```

Gets the list of task holiday information.

getTaskHolidayGroupInfoList

```
public OZMap getTaskHolidayGroupInfoList()
```

Gets the list of task holiday group information.

saveTaskHoliday

```
public void saveTaskHoliday()
```

Saves task holiday information as an xml file.

directExport

```
public DirectExportResult directExport(ServerInfo s, SortProperties configMap, SortProperties exportMap) throws SchedulerException
```

Export by applying the viewer parameter.

Parameters:

s - OZ Server information

configMap - OZ Scheduler configuration option

exportMap - Export information

directExportFiles

```
public Hashtable[] directExportFiles(ServerInfo s, SortProperties configMap,  
SortProperties exportMap, String localTempPath) throws SchedulerException
```

Export by applying the viewer parameter.

Parameters:

s - OZ Server information

configMap - OZ Scheduler configuration option

exportMap - Export information

localTempPath - local temporary path

directExportResult

```
public DirectExportResult directExportResult(ServerInfo s, SortProperties configMap,  
SortProperties exportMap)
```

Export by applying the viewer parameter.

Parameters:

s - OZ Server information

configMap - OZ Scheduler configuration option

exportMap - Export information

directExportByteArray

```
public HashTable directExportByteArray(ServerInfo s, SortProperties configMap,  
SortProperties exportMap) throws SchedulerException
```

Run the task.

Parameters:

s - OZ Server information

configMap - OZ Scheduler configuration option

exportMap - Export information

directPrint

```
public DirectPrintResult directPrint(ServerInfo s, SortProperties configMap,  
SortProperties printMap) throws SchedulerException
```

Print by applying the viewer parameter.

Parameters:

s - OZ Server information

configMap - Scheduler configuration information

printMap - Print information

ServerInfo

This is a class that contains information about OZ Server.

Method Detail

setURL

```
public final void setURL(String url) throws IllegalArgumentException
```

Sets the OZ Server URL.

Parameters:

url - OZ Server URL

setID

```
public final void setID(String id)
```

Sets the user ID.

Parameters:

id - User ID

setPWD

```
public final void setPWD(String pwd)
```

Sets the user password.

Parameters:

pwd - User password

TaskHolidayGroupInfo

This class defines a task holiday group.

Constructor Detail

TaskHolidayGroupInfo

```
public TaskHolidayGroupInfo(String name)
```

Defines the group of task holidays. The task holiday means the date when the task will not be performed.

Parameters:

name - Task holiday group name

Method Detail

getName

```
public String getName()
```

Gets the task holiday group name.

addReference

```
public void addReference(String name)
```

Registers the task holiday name that the task holiday group refers to.

Parameters:

name - Task holiday name

addReference

```
public void addReference(String[] names)
```

Registers the task holiday name that the task holiday group refers to.

Parameters:

names - The array of the task holiday name to be register in the task holiday group

getReferences

```
public String[] getReferences()
```

Gets the array of the reference name that will refer to the task holiday.

removeReference

```
public void removeReference(String name)
```

Deletes the task holiday name that the task holiday group refers to.

Parameters:

name - The task holiday name to be deleted in the task holiday group

removeReference

```
public void removeReference(String[] name)
```

Deletes the task holiday name that the task holiday group refers to.

Parameters:

name - The array of the task holiday name to be deleted in the task holiday group

removeAllRef

```
public void removeAllRef()
```

Deletes all task holiday names referred to by the task holiday group.

TaskHolidayInfo

This class defines a task holiday.

Constructor Detail

TaskHolidayInfo

```
public TaskHolidayInfo(String name)
```

Defines the task holiday. The task holiday means the date when the task will not be performed.

Parameters:

name - Task holiday name

Method Detail

getName

```
public String getName()
```

Gets task holiday name.

setBaseDay

```
public void setbaseDay(String pattern)
```

Set the base day to designate as task holiday.

Parameters:

pattern - Pattern of the baseday

getBaseDay

```
public String getBaseDay()
```

Gets the baseday designated as task holiday.

setStart

```
public void setStart(int start)
```

Sets how many days before the "baseday" will be holiday.

Parameters:

start - The number of days before the baseday.

getStart

```
public int getStart()
```

Gets the number of days before the baseday.

setEnd

```
public void setEnd(int end)
```

Sets how many days after the "baseday" will be holiday.

Parameters:

end - The number of days after the baseday.

getEnd

```
public void getEnd()
```

Gets the number of days after the baseday.

TaskResult

This class provides the results of the task execution.

Member Variables

taskID

```
public String taskID
```

ID of the task.

taskName

```
public String taskName
```

Name of the task.

taskGoupName

```
public String taskGoupName
```

Group name of the task.

completedTime

```
public String completedTime
```

Completion time of task.

isSuccessfulCode

`public int isSuccessfulCode`

Result code of task execution.(100001: failed, 200001: Export failed, 300001: Email sending failed, 400001: Succeeded)

isSuccessful

`public String isSuccessful`

Result of task execution.(Failed, Exporting Failed, Mailing Failed, Succeeded)

formCategoryName

`public String formCategoryName`

Category name of the OZR file.

formFileName

`public String formFileName`

Name of the OZR file.

Parameter

`public String Parameter`

Parameters set in the task.

schedulingType

`public String schedulingType`

Schedule type.(immediately, once, periodically)

exportFileList

`public String exportFileList`

List of exported file names. Multiple file names are listed with " | " as a separator.

errorMsg

`public String errorMsg`

Error message.

Interface

OZ Scheduler API provides the following Interface:

- [OZSchedulerUserDefinedPostTaskExecuteEx](#)

OZSchedulerUserDefinedPostTaskExecuteE

X

This interface provides a method to execute specific functions after successful completion of task execution.

Method Detail

postTaskExecuteSuccess

```
boolean postTaskExecuteSuccess(oz.util.SortProperties configMap,  
oz.util.SortProperties exportMap, OZUDEPostTarget target,  
org.apache.log4j.Category cat) throws OZSchedulerUDEException
```

If successful execution of the task, and then run the post-processing program to be executed next.

Parameters:

configMap - Config Properties

exportMap - Export Properties

target - Target

cat - Execution log

postTaskExecuteFail

```
boolean postTaskExecuteFail(oz.util.SortProperties configMap,  
oz.util.SortProperties exportMap, OZUDEPostTarget target,  
org.apache.log4j.Category cat) throws OZSchedulerUDEException
```

If failed execution of the task, and then run the post-processing program to be executed next.

Parameters:

configMap - Config Properties

exportMap - Export Properties

target - Target

cat - Execution log

OZ Repository API

OZ Repository API provides [classes](#) and [interfaces](#).

Class

OZ Repository API provides the following class:

- [RepositoryEX](#)

RepositoryEX

This class defines a task holiday group.

Constructor Detail

RepositoryEx

```
public RepositoryEx(String url, String id, String pw, boolean useUSL) throws  
OZCPEException
```

Performs repository functions such as group/user management function, category/item authority setting function, and item history management function.

Parameters:

url - The URL of OZ Server

id - The user ID

pw - The user password

useUSL - Whether to use the USL or not. Can be omitted. If omitted, false is applied.

Method Detail

The RepositoryEX class provides the following methods:

- Configuring the repository
- Creating groups and users
- Editing groups and users
- Deleting groups and users
- Getting groups and users
- Setting up group and user permissions
- Getting group and user permissions
- Creating categories and items
- Editing categories and items
- Deleting categories and items
- Getting categories and items
- Setting up category and item permissions
- Getting category and item permissions
- Using item version control

Configuring the repository

To configure the repository using the RepositoryEX class, use the method below:

Method Detail

getRepositoryConfig

```
public SortProperties getRepositoryConfig() throws OZCPEException
```

Gets the Repository configurations.

setRepositoryConfig

```
public void setRepositoryConfig(SortProperties prop) throws OZCPEException
```

Modifies the Repository configurations.

Parameters:

prop - Configurations to modify

reloadRepository

```
public void reloadRepository() throws OZCPEException
```

Reloads the Repository.

Creating groups and users

To create groups and users using the RepositoryEX class, use the method below:

Method Detail

createGroup

```
public String createGroup(String gName, String pGroupID, String desc) throws  
OZCPEException
```

Creates a new group and returns its ID.

Parameters:

gName - Group name

pGroupID - Parent group ID

desc - Group description

createUser

```
public String createUser(String userName, String password, String desc) throws  
OZCPEException
```

Creates a new user and returns its ID.

Parameters:

userName - User name

password - User password

desc - User description

createUserInGroup

```
public String createUserInGroup(String userName, String password, String groupID,  
String desc) throws OZCPException
```

Creates a new user in the designated group and returns the user ID.

Parameters:

userName - User name

password - User password

groupID - Group ID

desc - User description

Editing groups and users

To edit groups and users using the RepositoryEX class, use the method below:

Method Detail

modifyGroupName

```
public String modifyGroupName(String groupID, String gName) throws OZCPException
```

Modifies the name of the group corresponding to the designated user ID.

Parameters:

groupID - Group ID

gName - Group name

modifyGroupDesc

`public boolean modifyGroupDesc(String groupID, String desc) throws OZCPException`

Returns the group description corresponding to the designated group ID.

Parameters:

groupID - Group ID

desc - Group description

modifyUserName

`public String modifyUserName(String userID, String userName) throws OZCPException`

Modifies the user name corresponding to the designated user ID.

Parameters:

userID - User ID

userName - User name

modifyUserPassword

`public boolean modifyUserPassword(String userID, String old_password, String new_password) throws OZCPException`

Modifies the password of the user corresponding to the designated user ID.

Parameters:

userID - User ID

old_password - Old password

new_password - New password

modifyUserDesc

`public boolean modifyUserDesc(String userID, String desc) throws OZCPException`

Modifies the user description corresponding to the designated user ID.

Parameters:

userID - User ID

desc - User description

transferGroup

```
public boolean transferGroup(String groupID, String target_GroupID) throws  
OZCPEException
```

Transfers the group to another group and return success or failure.

Parameters:

groupID - Group ID to transfer

target_groupID - Target group ID

transferUser

```
public boolean transferUser(String userID, String target_GroupID) throws  
OZCPEException
```

Transfers the user to another group and return success or failure.

Parameters:

userID - User ID to transfer

target_groupID - Target group ID

Deleting groups and users

To delete groups and users using the RepositoryEX class, use the method below:

Method Detail

deleteGroup

```
public boolean deleteGroup(String groupID) throws OZCPException
```

Deletes the group corresponding to the designated group ID.

Parameters:

groupID - Group ID

deleteUser

```
public boolean deleteUser(String userID) throws OZCPException
```

Deletes the information of all users corresponding to the designated user ID.

Parameters:

userID - User ID

Getting groups and users

To get groups and users using the RepositoryEX class, use the method below:

Method Detail

getGroupInfo

```
public IOZGroupInfo getGroupInfo(String groupID) throws OZCPException
```

Gets the information of the group corresponding to the designated group ID.

Parameters:

groupID - Group ID

getGroupInfoListInGroup

```
public IOZGroupInfoList getGroupInfoListInGroup(String groupID, boolean  
isRecursive) throws OZCPException
```

Gets the group information that was added to the group.(Not supported in NONE type
repository)

Parameters:

groupID - Group ID

isRecursive - Whether to include sub-group

getSubGroupInfoList

```
public IOZGroupInfoList getSubGroupInfoList(String groupID) throws OZCPException
```

Gets the sub-group information of the designated group.

Parameters:

groupID - Group ID

getParentGroupInfo

```
public IOZGroupInfo getParentGroupInfo(String groupID) throws OZCPException
```

Gets the information of the parent group of the designated group.

Parameters:

groupID - Group ID

getUserInfo

```
public IOZUserInfo getUserInfo(String userID) throws OZCPException
```

Gets the information of the user corresponding to the designated user ID.

Parameters:

userID - User ID

getUserInfoList

```
public IOZUserInfoList getUserInfoList() throws OZCPException
```

Gets the information of all users.

getUserInfoListInGroup

```
public IOZUserInfoList getUserInfoListInGroup(String groupID) throws OZCPException
```

Gets the information of all users registered in the designated group ID.

Parameters:

groupID - Group ID

getUserInfoListInGroup

```
public IOZUserInfoList getUserInfoListInGroup(String groupID, boolean isRecursive)
```

```
throws OZCPException
```

Gets the user information that was added to the group. (Not supported in NONE type repository)

Parameters:

groupID - Group ID

isRecursive - Whether to include sub-group user

checkUserPassword

```
public boolean checkUserPassword(String userID, String password) throws
```

```
OZCPException
```

Checks whether the user password is correct.

Parameters:

userID - User ID

password - User password

getGroupIdOfUser

```
public String getGroupIdOfUser(String userID) throws OZCPException
```

Gets the information of the group where the designated user belongs.

Parameters:

userID - User ID

Setting up group and user permissions

To set group and user permissions using the RepositoryEX class, use the method below:

Method Detail

addGroupAdmin

```
public boolean addGroupAdmin(Object sessionID, String userID, String groupID)  
throws OZRepositoryException
```

Adds group administrator to the designated group.

Parameters:

sessionID - Session ID

userID - User ID

groupID - Group ID

removeGroupAdmin

```
public boolean removeGroupAdmin(String userID, String groupID) throws  
OZCPEException
```

Cancel group administrator of the designated group.

Parameters:

userID - User ID

groupID - Group ID

addAdmin

```
public boolean addAdmin(String uID) throws OZCPEException
```

Adds administrator to the designated user ID.(Not supported in NONE type repository)

Parameters:

uID - User ID

removeAdmin

```
public boolean removeAdmin(String uID) throws OZCPEException
```

Cancel administrator of the designated user ID.(Not supported in NONE type repository)

Parameters:

uID - User ID

getUserAdminListInGroup

```
public IOZUserInfoList getUserAdminListInGroup(String groupID) throws
```

```
OZCPEException
```

Gets the group administrator information of the designated group.

Parameters:

groupID - Group ID

EnableLogin

```
public void EnableLogin(String userID)
```

Sets the enable the login of the user corresponding to the designated user ID.

Parameters:

userID - User ID

DisableLogin

```
public void DisableLogin(String userID)
```

Set to disable the login of the user corresponding to the designated user ID.

Parameters:

userID - User ID

enableUserLogin

```
public void enableUserLogin(String userID) throws OZCPException
```

Set to enable the login of the user corresponding to the designated user ID.

Parameters:

userID - User ID

disableUserLogin

```
public void disableUserLogin(String userID) throws OZCPException
```

Set to disable the login of the user corresponding to the designated user ID.

Parameters:

userID - User ID

isUserGroupAdmin

```
public boolean isUserGroupAdmin(String userID, String groupID) throws  
OZCPException
```

Checks that the user corresponding to the designated user ID is the administrator of the group.

Parameters:

userID - User ID

groupID - Group ID

isCheckAdmin

```
public boolean isCheckAdmin(String userID) throws OZCPException
```

Checks that the user corresponding to the designated user ID is the administrator of the Repository.

Parameters:

userID - User ID

isLoginEnabled

```
public boolean isLoginEnabled(String userID) throws OZCPEException
```

Gets whether the login of the user corresponding to the designated user ID.

Parameters:

userID - User ID

Getting group and user permissions

To get group and user permissions using the RepositoryEX class, use the method below:

Method Detail

getGroupAuthToCategory

```
public int getGroupAuthToCategory(String categoryID, String groupID) throws  
OZCPEException
```

Gets the permission of the given group for the given category.

Parameters:

categoryID - Category ID

groupID - Group ID

getGroupListAuthToCategory

```
public IOZGroupInfoList getGroupListAuthToCategory(String categoryID, int perm)  
throws OZCPEException
```

Gets the information of all users with permission level of or higher than Perm for the designated category ID.

Parameters:

categoryID - Category ID
perm - Permission information

getGroupAuthToItem

```
public int getGroupAuthToItem(String groupID, String itemID) throws OZCPException
```

Gets the permission of the given group ID for the given item ID.

Parameters:

groupID - Group ID
itemID - Item ID

getGroupListAuthToItem

```
public IOZGroupInfoList getGroupListAuthToItem(String itemID, int perm) throws OZCPException
```

Gets the information of all groups with permission level of or higher than Perm for the item ID.

Parameters:

itemID - Item ID
perm - Permission information

getUserAuthToCategory

```
public int getUserAuthToCategory(String userID, String categoryID) throws OZCPException
```

Gets the permission of the given user ID for the given category ID.

Parameters:

userID - User ID

categoryID - Category ID

getUserListAuthToCategory

```
public IOZUserInfoList getUserListAuthToCategory(String categoryID, int perm)  
throws OZCPEException
```

Gets the list of all users with permission level of or higher than Perm for the designated category ID.

Parameters:

categoryID - Category ID

perm - Permission information

getUserAuthToItem

```
public int getUserAuthToItem(String userID, String itemID) throws OZCPEException
```

Gets the user permission for the given item ID

Parameters:

userID - User ID

itemID - Item ID

perm - Permission information

getUserListAuthToItem

```
public IOZUserInfoList getUserListAuthToItem(String itemID, int perm) throws  
OZCPEException
```

Gets the list of all items for which the user's permission level is or higher than Perm.

Parameters:

itemID - Item ID

perm - Permission information

Creating categories and items

To create categories and items using the RepositoryEX class, use the method below:

Method Detail

createCategory

```
public String[] createCategory(String[] categoryNames, String[] pCategoryIDs,  
String comment, int[] errorCodes, String[] errorMsg) throws OZCPException
```

Creates a new category and returns its ID.

Parameters:

categoryName - Category name

pCategoryIDs - Parent category ID

comment - Comment

errorCode - Error code

errorMsg - Error message

createItem

```
public String[] createItem(String[] itemNames, String[] itemDescs, String[]  
categoryIDs, boolean[] isComp, InputStream[] itemStreams, String comment, int[]  
errorCodes, String[] errorMsg) throws OZCPException
```

Creates an item and returns item ID.

Parameters:

itemNames - Item Name

itemDescs - Item description

categoryIDs - Category ID

isComp - Whether to compress the item

itemStreams - Input stream

comment - Comment

errorCodes - Error code

errorMsg - Error message

createItem

```
public String[] createItem(String[] itemNames, String[] itemDescs, String[] categoryIDs, boolean[] isComp, InputStream[] itemStreams, String comment, long[] createdTimes, int[] errorCodes, String[] errorMsg) throws OZCPException
```

Creates an item by specifying the time, and returns item ID.(Not supported in NONE type repository)

Parameters:

itemNames - Item Name

itemDescs - Item description

categoryIDs - Category ID

isComp - Whether to compress the item

itemStreams - Input stream

comment - Comment

createdTimes - Time the item was created. If the value is set to 0, it is applied to the current time.

errorCodes - Error code

errorMsg - Error message

Editing categories and items

To edit categories and items using the RepositoryEX class, use the method below:

Method Detail

modifyCategoryName

```
public String modifyCategoryName(String categoryID, String new_CategoryName,  
String comment) throws OZCPEException
```

Modifies the name of the category corresponding to the designated category ID.

Parameters:

categoryID - Category ID

new_categoryName - Category name

comment - Comment

modifyCategoryDesc

```
public boolean modifyCategoryDesc(String categoryID, String categoryDesc) throws  
OZCPEException
```

Modifies the category description. If the repository type is NONE, it only works if

REPOSITORY_NONE_BACKUP=true.

Parameters:

categoryID - Category ID

categoryDesc - Category description

modifyItemName

```
public String modifyItemName(String itemID, String itemName, String comment)  
throws OZCPEException
```

Modifies an item name and returns its ID.

Parameters:

itemID - Item ID

new_itemName - Item name

comment - Comment

modifyItemDesc

```
public boolean modifyItemDesc(String itemID, String iDesc) throws OZCPException
```

Modifies the description of the item corresponding to the item ID.

Parameters:

itemID - Item ID

desc - Item description

modifyItemDatetime

```
public boolean modifyItemDatetime(String itemID, long datetime) throws  
OZCPException
```

Modifies the creation time of the item.(Not supported in NONE type repository)

Parameters:

itemID - Item ID

datetime - Time the item was changed. If the value is set to 0, it is applied to the current time.

transferCategory

```
public boolean transferCategory(String categoryID, String target_CategoryID)  
throws OZCPException
```

Transfers the category to another category and return success or failure.

Parameters:

categoryID - Category ID

target_CategoryID - Target Category ID

transferItem

```
public boolean transferItem(String[] itemIDs, String target_CategoryID) throws  
OZCPException
```

Transfers the item to another category and return success or failure.

Parameters:

itemIDs - Item ID

target_CategoryID - Target category ID

encryptFile

```
public Vector encryptFile(String[] itemIDs) throws OZCPEException
```

Encrypts the OZR files and returns a list of files that failed to encrypt.

Parameters:

itemIDs - Item ID

encryptFile

```
public Vector encryptFile(String categoryID, boolean isRecursive) throws  
OZCPEException
```

Encrypts the OZR files and returns a list of files that failed to encrypt.

Parameters:

categoryID - Category ID

isRecursive - Whether to include sub-categories

decryptFile

```
public Vector decryptFile(String[] itemIDs) throws OZCPEException
```

Decrypts the OZR files and returns a list of files that failed to decrypt.

Parameters:

itemIDs - Item ID

decryptFile

```
public Vector decryptFile(String categoryID, boolean isRecursive) throws  
OZCPEException
```

Decrypts the OZR files and returns a list of files that failed to decrypt.

Parameters:

categoryID - Category ID

isRecursive - Whether to include sub-categories

setUserDefinedResorceSync

```
public int setUserDefinedResourceSync(String alias, String[] itemIDs, String  
comment, String[] errorMsg) throws OZCPException
```

Synchronize OZR files with the user defined resource and returns the result.

Parameters:

alias - Alias defined in ozudr.properties file

itemIDs - Full path of the report

Comment - comments

errorMsg - Error message

Deleting categories and items

To delete categories and items using the RepositoryEX class, use the method below:

Method Detail

deleteCategory

```
public boolean[] deleteCategory(String[] categoryIDs, boolean[] isDestroys, String  
comment, int[] errorCodes, String[] errorMsg) throws OZCPException
```

Deletes the category corresponding to the designated category ID.

Parameters:

categoryIDs - Category ID

isDestroys - Whether to permanently delete the category

comment - Comment

errorCode - Error code

errorMsg - Error message

unDeleteCategory

```
public boolean[] unDeleteCategory(String[] categoryIDs, String comment, int[]
errorCodes, String[] errorMsg) throws OZCPException
```

Restores a deleted category.

Parameters:

categoryIDs - Category ID

comment - Comment

errorCode - Error code

errorMsg - Error message

deleteItem

```
public boolean[] deleteItem(String[] itemIDs, boolean[] isDestroys, String
comment, int[] errorCodes, String[] errorMsg) throws OZCPException
```

Deletes the item from the Repository.

Parameters:

itemIDs - Item ID

isDestroys - Whether to permanently delete the item

comment - Comment

errCodes - Error code

errMsgs - Error message

unDeleteItem

```
public boolean[] unDeleteItem(String[] itemIDs, String comment, int[] errorCodes,  
String[] errorMsg) throws OZCPException
```

Restores the deleted item.

Parameters:

itemIDs - Item ID

comment - Comment

errCodes - Error code

errMsgs - Error message

Getting categories and items

To get categories and items using the RepositoryEX class, use the method below:

Method Detail

getCategoryInfo

```
public IOZCategoryInfo getCategoryInfo(String categoryId) throws OZCPException
```

Gets the information of the designated category.

Parameters:

categoryId - Category ID

getCategoryListInCategory

```
public IOZCategoryInfoList getCategoryListInCategory(String categoryId) throws  
OZCPException
```

Gets the information of all sub categories of the designated category.

Parameters:

categoryID - Category ID

getCategoryIdOfItem

```
public String getCategoryIdOfItem(String itemID) throws OZCPEException
```

Returns the ID of the category where the item exist.

Parameters:

itemID - Item ID

getDeletedItemListInCategory

```
public IOZItemInfoList getDeletedItemListInCategory(String categoryID) throws  
OZCPEException
```

Gets the information of deleted items in the designated category.

Parameters:

categoryID - Category ID

getItems

```
public InputStream[] getItems(String[] itemIDs, long[] modifiedTimes, boolean[]  
isComp, boolean[] isObjStream, String[] categoryIDs, int[] errorCodes, String[]  
errorMsg) throws OZCPEException
```

Gets the item corresponding to the item ID.

Parameters:

itemIDs - Item ID

modifiedTimes - Last update time of the item

isComp - Whether to compress it

isObjStream - Whether the ODI object stream is necessary

errCodes - Error code

errMsgs - Error message

getItemInfo

```
public IOZItemInfo getItemInfo(String itemID) throws OZCPException
```

Gets the Information of the item corresponding to the item ID.

Parameters:

itemID - Item ID

getItemInfoListInCategory

```
public IOZItemInfoList getItemInfoListInCategory(String categoryID, boolean  
isRecursive) throws OZCPException
```

Gets the information of the item.(Not supported in NONE type repository)

Parameters:

categoryID - Category ID

isRecursive - Whether to include sub-categories

getItemsUnCondition

```
public InputStream[] getItemsUnCondition(String[] itemIDs, boolean[] isComp,  
boolean[] isObjStream, String[] categoryIDs, int[] errorCodes, String[] errorMsg)  
throws OZCPException
```

Gets the itemID without conditions.

Parameters:

itemIDs - Item ID

isComp - Whether to compress it

isObjStream - Whether the ODI object stream is necessary

errCodes - Error code

errMsgs - Error message

getItemCountInCategory

```
public int getItemCountInCategory(String categoryID) throws OZCPException
```

Gets the number of all items in the designated category.

Parameters:

categoryID - Category ID

getItemListInCategory

```
public IOZItemInfoList getItemListInCategory(String categoryID) throws  
OZCPException
```

Gets the information of all items in the designated category.

Parameters:

categoryID - Category ID

hasTheItem

```
public boolean hasTheItem(String itemID) throws OZCPException
```

Gets the Information of the item corresponding to the item ID.

Parameters:

itemID - Item ID

Setting up category and item permissions

To set category and item permissions using the RepositoryEX class, use the method below:

Method Detail

modifyCategoryGroupAuth

```
public boolean modifyCategoryGroupAuth(String categoryID, String groupID, int perm) throws OZCPEException
```

Modify permission of the given category

Parameters:

categoryID - Category ID

groupID - Group ID

perm - Permission information

modifyCategoryUserAuth

```
public boolean modifyCategoryUserAuth(String userID, String categoryID, int perm) throws OZCPEException
```

Modify the permission of the given user ID for the given category ID.

Parameters:

userID - User ID

categoryID - Category ID

perm - Permission information

modifyItemGroupAuth

```
public boolean modifyItemGroupAuth(String groupID, String itemID, int perm) throws OZCPEException
```

Modify the permission of the given group ID for the given item ID.

Parameters:

groupID - Group ID

itemID - Item ID

perm - Permission information

modifyItemUserAuth

```
public boolean modifyItemUserAuth(String userID, String itemID, int perm) throws  
OZCPEException
```

Modify the user permission for the given item ID.

Parameters:

userID - User ID

itemID - Item ID

perm - Permission information

Getting category and item permissions

To get category and item permissions using the RepositoryEX class, use the method below:

Method Detail

getCategoryListAuthToGroupInCategory

```
public IOZCategoryInfoList getCategoryListAuthToGroupInCategory(String groupID,  
String categoryID, int perm) throws OZCPEException
```

Gets the information of all categories with permission level of or higher than Perm for the group corresponding to the designated group ID.

Parameters:

groupID - Group ID

categoryID - Category ID

perm - Permission information

getCategoryListAuthToUserInCategory

```
public IOZCategoryInfoList getCategoryListAuthToUserInCategory(String userID,  
String categoryID, int perm) throws OZCPEException
```

Gets the list of all categories for which the designated user's permission level is or higher than Perm among the sub categories in the designated category.

Parameters:

userID - User ID

categoryID - Category ID

perm - Permission information

getItemListAuthToGroup

```
public IOZItemInfoList getItemListAuthToGroup(String groupID, int perm) throws  
OZCPEException
```

Gets the list of all items with permission level of or higher than Perm for the group corresponding to the designated group ID.

Parameters:

groupID - Group ID

perm - Permission information

getItemListAuthToGroupInCategory

```
public IOZItemInfoList getItemListAuthToGroupInCategory(String groupID, String  
categoryID, int perm) throws OZCPEException
```

Gets the information of all items with permission level of or higher than Perm for the designated category ID.

Parameters:

groupID - Group ID

categoryID - Category ID

perm - Permission information

getItemListAuthToUser

```
public IOZItemInfoList getItemListAuthToUser(String userID, int perm) throws  
OZCPException
```

Gets the list of all items for which the user's permission level is or higher than Perm among the items in the designated category.

Parameters:

userID - User ID

perm - Permission information

getItemListAuthToUserInCategory

```
public IOZItemInfoList getItemListAuthToUserInCategory(Object sessionID, String  
userID, String categoryID, int perm) throws OZRepositoryException
```

Gets the list of all items for which the user has permission level of or higher than Perm among the items in the designated category.

Parameters:

sessionID - Session ID

userID - User ID

categoryID - Category ID

perm - Permission information

Using item version control

To use item version control function using the RepositoryEX class, use the method below:

Method Detail

checkOutItem

```
public InputStream[] checkOutItem(String itemIDs[], String[] localCheckOutFolders,
long[] localFileTimes, boolean[] isComp, int[] errorCodes, String[] errorMsg)
throws OZCPException
```

Check out the item to a folder.

Parameters:

itemIDs - Item ID

localCheckOutFolders - Local folder name

localFileTimes - Local file time

isComp - Whether to compress it

errorCodes - Error code

errorMsg - Error message

checkOutItem

```
public InputStream[] checkOutItem(String itemIDs[], String[] localCheckOutFolders,
long[] localFileTimes, boolean[] isComp, String[] checkOutCmts, int[] errorCodes,
String[] errorMsg) throws OZCPException
```

Check out the item to a folder.

Parameters:

itemIDs - Item ID

localCheckOutFolders - Local folder name

localFileTimes - Local file time

isComp - Whether to compress it

checkOutCmts - Comment

errorCodes - Error code

errorMsg - Error message

checkInItem

```
public boolean[] checkInItem(String[] itemIDs, boolean[] isComp, InputStream[]
iStreams, String comment, boolean[] keepCheckOut, int[] errorCodes, String[]
errorMsg) throws OZCPException
```

Check in the item.(Not supported in NONE type repository)

Parameters:

itemIDs - Item ID

isComp - Whether to compress it

iStreams - Input stream of the item

comment - Comment

keepCheckOut - Whether to keep the check out status

errorCodes - Error code

errorMsg - Error message

checkInItem

```
public boolean[] checkInItem(String[] itemIDs, boolean[] isComp, InputStream[]
itemStreams, String comment, boolean[] keepCheckOut, long[] datetimes, int[]
errorCodes, String[] errorMsg) throws OZCPException
```

Check in the item by specifying the time.(Not supported in NONE type repository)

Parameters:

itemIDs - Item ID

isComp - Whether to compress it

itemStreams - Input stream of the item

comment - Comment

keepCheckOut - Whether to keep the check out status

datetimes - Time the item was checked in. If the value is set to 0, it is applied to the current time.

errorCodes - Error code

errorMsg - Error message

undoCheckOutItem

```
public InputStream[] undoCheckOutItem(String[] itemIDs, boolean[] isReplaces,  
boolean[] isComp, int[] errorCodes, String[] errorMsg) throws OZCPException
```

Undo check out the item.

Parameters:

itemIDs - Item ID

isReplaces - Whether to get the items in the local working folder

isComp - Whether to compress it

errCodes - Error code

errMsgs - Error message

isCheckOutUser

```
public boolean[] isCheckOutUser(String[] itemIDs) throws OZCPException
```

Checks that the current user checked out the item.

Parameters:

itemIDs - Item ID

getItemByVersion

```
public InputStream getItemByVersion(String itemID, int version, boolean isComp)  
throws OZCPException
```

Gets the item of the designated version among the items corresponding to the item ID.

Parameters:

itemID Item ID

version - Item version

isComp - Whether to compress it

getHistoryItemList

```
public IOZHistoryInfoList getHistoryItemList(String itemID) throws OZCPException
```

Gets the history information of the item.

Parameters:

itemID - Item ID

getHistoryItemListByDatetime

```
public IOZHistoryInfoList getHistoryItemListByDatetime(String itemID) throws  
OZCPException
```

Gets the history information of the item the order date.(Not supported in NONE type repository)

Parameters:

itemID - Item ID

getHistoryItemByDatetime

```
public InputStream getHistoryItemByDatetime(String itemID, long datetime, boolean  
isComp, boolean isObjStream) throws OZCPException
```

Gets Most Recent Items specified time.(Not supported in NONE type repository)

Parameters:

itemID - Item ID

datetime - Time the item was checked in

isComp - Whether to compress it

isObjStream - Whether the obj item is objstream

getDeleteHistoryItemInfo

```
public IOZHistoryInfoList getDeleteHistoryItemInfo(String itemID) throws  
OZCPException
```

Gets the information of a deleted item.

Parameters:

itemID - Item ID

modifyHistoryItemComment

```
public boolean modifyHistoryItemComment(String itemID, int itemVersion, String  
newComment) throws OZCPException
```

Modifies the comment of the specified version.

Parameters:

itemID - Item ID

itemversion - Item version

newComment - Comment

modifyHistoryItemDatetime

```
public boolean modifyHistoryItemDatetime(String itemID, int itemVersion, long  
datetime) throws OZCPException
```

Modifies the item time of the specified version.(Not supported in NONE type repository)

Parameters:

itemID - Item ID

itemVersion - Item version

datetime - Time the item was checked in. If the value is set to 0, it is applied to the current time.

removeHistoryItem

```
public boolean removeHistoryItem(String itemID, int itemVersion) throws  
OZCPException
```

Gets the item's specific version of history.

Parameters:

itemID - Item ID

itemversion - Item version

rollBackItem

```
public boolean rollBackItem(String itemID, int itemVersion, String comment) throws  
OZCPException
```

Rolls back the item to the designated version.

Parameters:

itemID - Item ID

itemVersion - Item version

comment - Comment

Interface

OZ Repository API provides the following Interfaces:

- [IOZCategoryInfo](#)
- [IOZCategoryInfoList](#)
- [IOZGroupInfo](#)
- [IOZGroupInfoList](#)
- [IOZHistoryInfo](#)
- [IOZHistoryInfoList](#)
- [IOZItemInfo](#)
- [IOZLoginInfo](#)
- [IOZRepository](#)
- [IOZRepositoryAuthGroupCategory](#)
- [IOZRepositoryAuthGroupItem](#)
- [IOZRepositoryAuthUserCategory](#)
- [IOZRepositoryAuthUserItem](#)
- [IOZRepositoryCategory](#)
- [IOZRepositoryExternalLogin](#)
- [IOZRepositoryGroup](#)
- [IOZRepositoryItem](#)
- [IOZRepositoryItemHistory](#)
- [IOZRepositoryMultiLoginUser](#)
- [IOZRepositoryUser](#)
- [IOZUserInfo](#)
- [IOZUserInfoList](#)

IOZCategoryInfo

This interface provides category information.

Method Detail

getCategoryID

```
public String getCategoryID()
```

Gets the ID of the current category.

getCategoryName

```
public String getCategoryName()
```

Gets the name of the current category.

getParentCategoryID

```
public String getParentCategoryID()
```

Gets the parent category ID of the current category.

getParentCategoryName

```
public String getParentCategoryName()
```

Gets the parent category name of the current category.

getComment

```
public String getComment()
```

Gets the comment of the current category.

getDesc

```
public String getDesc()
```

Gets the description of the current category.

getUpdateTime

```
public long getUpdateTime()
```

Gets the update time of the current category.

getCreateUserID

```
public String getCreateUserID()
```

Gets the user ID that created the current category.

getCreateUserName

```
public String getCreateUserName()
```

Gets the user name that created the current category.

getPerm

```
public int getPerm()
```

Gets the permission information of the current category.

IOZCategoryInfoList

This interface provides category information.

Method Detail

getCategoryInfo

```
public IOZCategoryInfo getCategoryInfo(int index)
```

Gets the category information corresponding to the index.

Parameters:

index - Index

getSize

```
public String getSize()
```

Gets the number of the category lists.

getCategoryInfos

```
public IOZCategoryInfo[] getCategoryInfos()
```

Gets the category information as array.

getItemInfo

```
public IOZItemInfo getItemInfo(int index)
```

Gets the information of the item corresponding to the index.

Parameters:

index - Index

getSize

```
public String getSize()
```

Gets the number of the item lists.

getItemInfos

```
public IOZItemInfo[] getItemInfos()
```

Gets the item information as array.

IOZGroupInfo

This interface provides group information.

Method Detail

isGroupAdmin

```
public boolean isGroupAdmin(String userID)
```

Gets whether the designated user ID is the administrator of the group.

Parameters:

userID - User ID

getGroupAdminList

```
public HashMap getGroupAdminList()
```

Gets the GroupAdminList of the group.

getGroupID

```
public String getGroupID()
```

Gets the ID of the group.

getGroupName

```
public String getGroupName()
```

Gets the name of the group.

getParentGroupID

```
public String getParentGroupID()
```

Gets the parent group ID of the group.

getParentGroupName

```
public String getParentGroupName()
```

Gets the parent group name of the group.

getDesc

```
public String getDesc()
```

Gets the description of the group.

getUpdateTime

```
public long getUpdateTime()
```

Gets the update time of the group.

IOZGroupInfoList

This interface provides group information.

Method Detail

getGroupInfo

```
public IOZGroupInfo getGroupInfo(int index)
```

Gets the group information corresponding to the index.

Parameters:

index - Index

getSize

```
public String getSize()
```

Gets the number of group lists.

getGroupInfos

```
public IOZGroupInfo[] getGroupInfos()
```

Gets the group information as array.

IOZHistoryInfo

This interface provides information on the item's change history.

Method Detail

getItemID

```
public String getItemID()
```

Gets an item ID.

getItemName

```
public String getItemName()
```

Gets an item name.

getHistoryItemVersion

```
public int getHistoryItemVersion()
```

Gets the version count of an item.

getHistoryCheckInTime

```
public long getHistoryCheckInTime()
```

Gets the check in time of an item by version.

getHistoryCheckInUser

```
public String getHistoryCheckInUser()
```

Gets the check in user ID of the item by version.

getHistoryCheckInComment

```
public String getHistoryCheckInComment()
```

Gets the check in comment of the item by version.

IOZHistoryInfoList

This interface provides information on the item's change history.

Method Detail

getHistoryInfo

```
public IOZHistoryInfo getHistoryInfo(int index)
```

Gets the history information corresponding to the index.

Parameters:

index - Index

getSize

```
public String getSize()
```

Gets the number of history lists.

getHistoryInfos

```
public IOZHistoryInfo[] getHistoryInfos()
```

Gets the history information as array.

IOZItemInfo

This interface provides item information.

Method Detail

isCheckOut

```
public boolean isCheckOut()
```

Gets whether the item is checked out.

isDeletable

```
public boolean isDeletable()
```

Gets whether the item is deleted.

getItemID

```
public String getItemID()
```

Gets the item ID of the item.

getItemName

```
public String getItemName()
```

Gets the item name of the item.

getCategoryID

```
public String getCategoryID()
```

Gets the ID of the category where the item belongs.

getCategoryName

```
public String getCategoryName()
```

Gets the name of the category where the item belongs.

getUpdateTime

```
public long getUpdateTime()
```

Gets the update time of the item.

getDesc

```
public String getDesc()
```

Gets the description of the item.

getComment

```
public String getComment()
```

Gets the comment of the item.

getCreateUserID

```
public String getCreateUserID()
```

Gets the user ID that created the item.

getCreateUserName

```
public String getCreateUserName()
```

Gets the user name that created the item.

getCheckOutUser

```
public String getCheckOutUser()
```

Gets the user name that checked out the item.

getCheckOutLocalPath

```
public String getCheckOutLocalPath()
```

Gets the folder name of the checked-out item.

getPerm

```
public int getPerm()
```

Gets the permission information of the item.

IOZLoginInfo

This interface provides the login information.

Method Detail

getViewerType

public int getViewerType()

Gets the type of the viewer called by the client.

getMACAddress

public String getMACAddress()

Gets the MAC Address from the client.

getIPAddress

public String getIPAddress()

Gets the IP Address from the client.

getHDDSerialNo

public String getHDDSerialNo()

Gets the HDD Serial number from the client.

getUserID

public String getUserID()

Gets the user ID that the client logged in.

getUserName

public String getUserName()

Gets the user name that the client logged in.

getSessionID

public String getSessionID()

Gets the session ID created when the client logged in.

getClientIP

public String getClientIP()

Gets the client IP requested by the client.

IOZRepository

This interface provides repository setup functions.

Method Detail

setProperty

```
public void setProperty(SortProperties prop) throws OZRepositoryException
```

Modifies the repository configurations. Values set in the "repository.properties" are allowed to be modified.

Parameters:

prop - Repository configuration property

getRepositoryVersion

```
public int getRepositoryVersion()
```

Defines the Repository version and gets the defined version.

openRepository

```
public void openRepository() throws OZRepositoryException
```

Opens the repository when running the server.

closeRepository

```
public void closeRepository() throws OZRepositoryException
```

Closes the repository when exiting the server.

checkRepositoryLogin

```
public Object checkRepositoryLogin(String userName, String password, IOZLoginInfo  
loginInfo, HttpServletRequest request, HttpServlet servlet) throws  
OZRepositoryException
```

Gets the session information that logged in by calling the corresponding method when calling the client.

Parameters:

userName - Name of the logged in user

password - Password of the currently logged in user

loginInfo - Log in information

request - HTTP Servelt Request

servlet - HTTP Servlet

updateSessionState

```
public void updateSessionState(Object sessionID, int type) throws  
OZRepositoryException
```

Gets the session information that logged in by calling the corresponding method when calling the client.

Parameters:

sessionID - Session ID

type - Server status

repositoryLogout

```
public boolean repositoryLogout(Object sessionID, String userName, IOZLoginInfo  
loginInfo, HttpServletRequest request, HttpServlet servlet) throws  
OZRepositoryException
```

Log out, or release the user corresponding to the designated user ID and gets the result.

Parameters:

sessionId - Session ID

userName - User name

loginInfo - Log in information(Refer to the corresponding interface for details)

request - HTTP Servlet Request

servlet - HTTP Servlet

getRepositoryCategory

```
public IOZRepositoryCategory getRepositoryCategory() throws OZRepositoryException
```

Gets the class where the Category Interface is implemented.

getRepositoryItem

```
public IOZRepositoryItem getRepositoryItem() throws OZRepositoryException
```

Gets the class where the Item Interface is implemented.

getRepositoryItemHistory

```
public IOZRepositoryItemHistory getRepositoryItemHistory() throws  
OZRepositoryException
```

Gets the class where the Item History Interface is implemented.

getRepositoryGroup

```
public IOZRepositoryGroup getRepositoryGroup() throws OZRepositoryException
```

Gets the class where the Group Interface is implemented.

getRepositoryUser

```
public IOZRepositoryUser getRepositoryUser() throws OZRepositoryException
```

Gets the class where the User Interface is implemented.

getRepositoryMultiLoginUser

```
public IOZRepositoryMultiLoginUser getRepositoryMultiLoginUser() throws  
OZRepositoryException
```

Gets the class where the MultiLogin User Interface is implemented.

getRepositoryAuthGroupCategory

```
public IOZRepositoryAuthGroupCategory getRepositoryAuthGroupCategory() throws  
OZRepositoryException
```

Gets the class where the AuthGroupCategory Interface.

getRepositoryAuthGroupItem

```
public IOZRepositoryAuthGroupItem getRepositoryAuthGroupItem() throws  
OZRepositoryException
```

Gets the class where the AuthGroupItem Interface is implemented.

getRepositoryAuthUserCategory

```
public IOZRepositoryAuthUserCategory getRepositoryAuthUserCategory() throws  
OZRepositoryException
```

Gets the class where the AuthUserCategory Interface is implemented.

getRepositoryAuthUserItem

```
public IOZRepositoryAuthUserItem getRepositoryAuthUserItem() throws  
OZRepositoryException
```

Gets the class where the AuthUserItem Interface is implemented.

IOZRepositoryAuthGroupCategory

This interface provides functions to set category permissions per group.

Method Detail

getAccessTypeAuthGroupCategory

```
public int getAccessTypeAuthGroupCategory()
```

Gets the methods with access to the AuthGroupCategory Interface as numbers.

- int ACCESS_TYPE_AUTH_GROUP_CATEGORY_NOT = 0x00000000
- int ACCESS_TYPE MODIFY_CATEGORY_GROUPAUTH = 0x00000001
- int ACCESS_TYPE_GET_GROUPAUTH_TO_CATEGORY = 0x00000002
- int ACCESS_TYPE_GET_GROUPLIST_AUTH_TO_CATEGORY = 0x00000004
- int ACCESS_TYPE_GET_ITEMLIST_AUTH_TO_GROUP_IN_CATEGORY = 0x00000008
- int ACCESS_TYPE_GET_CATEGORYLIST_AUTH_TO_GROUP_IN_CATEGORY = 0x00010000

getOZRepository

```
public IOZRepository getOZRepository() throws OZRepositoryException
```

Gets the class where the OZRepository Interface class is implemented.

modifyCategoryGroupAuth

```
public boolean modifyCategoryGroupAuth(Object sessionID, String categoryID, String groupID, int perm) throws OZRepositoryException
```

Modifies the Permission information of the designated category and group.

Parameters:

sessionID - Session ID

categoryID - Category ID to change its permission

groupID - Group ID to change its permission

perm - Permission information(1: VIEW, 2: READ, 4: WRITE)

getGroupAuthToCategory

```
public int getGroupAuthToCategory(Object sessionID, String categoryID, String groupID) throws OZRepositoryException
```

Gets permission of given group for given category.

Parameters:

sessionID - Session ID

categoryID - Category ID to get its permission

groupID - Group ID

getGroupListAuthToCategory

```
public IOZGroupInfoList getGroupListAuthToCategory(Object sessionID, String categoryID, int perm) throws OZRepositoryException
```

Gets the user information of all users with permission level of or higher than Perm for the designated category ID.

Parameters:

sessionID - Session ID

categoryID - Category ID

perm - Permission information

getItemListAuthToGroupInCategory

```
public IOZItemInfoList getItemListAuthToGroupInCategory(Object sessionID, String groupID, String categoryID, int perm) throws OZRepositoryException
```

Gets the list of all items in the designated category with permission level of or higher than Perm for the designated group.

Parameters:

sessionID - Session ID

groupID - Group ID

categoryID - Category ID

perm - Permission information

getCategoryListAuthToGroupInCategory

```
public IOZCategoryInfoList getCategoryListAuthToGroupInCategory(Object sessionID, String groupID, String categoryID, int perm) throws OZRepositoryException
```

Gets the list of all categories among the sub categories in the designated category with permission level of or higher than Perm for the designated group.

Parameters:

sessionID - Session ID

groupID - Group ID

categoryID - Category ID

perm - Permission information

IOZRepositoryAuthGroupItem

This interface provides functions to set item permissions per group.

Method Detail

getAccessTypeAuthGroupItem

```
public int getAccessTypeAuthGroupItem()
```

Gets the methods with access to the AuthGroupItem Interface as numbers.

- int ACCESS_TYPE_AUTH_GROUP_ITEM_NOT = 0x00000000
- int ACCESS_TYPE MODIFY ITEM GROUPAUTH = 0x00000010
- int ACCESS_TYPE GET GROUPAUTH TO ITEM = 0x00000020
- int ACCESS_TYPE GET GROUPLIST AUTH TO ITEM = 0x00000040
- int ACCESS_TYPE GET ITEMLIST AUTH TO GROUP = 0x00000080

getOZRepository

```
public IOZRepository getOZRepository() throws OZRepositoryException
```

Gets the class where the OZRepository Interface is implemented.

modifyItemGroupAuth

```
public boolean modifyItemGroupAuth(Object sessionID, String groupID, String itemID, int perm) throws OZRepositoryException
```

Modifies the permission information of the designated group ID for the item ID.

Parameters:

sessionID - Session ID

groupID - Group ID

itemID - Item ID

perm - Permission information

getGroupAuthToItem

```
public int getGroupAuthToItem(Object sessionID, String groupID, String itemID)
throws OZRepositoryException
```

Gets the permission information of the designated group ID for the item ID.

Parameters:

sessionID - Session ID

groupID - Group ID

itemID - Item ID

getGroupListAuthToItem

```
public IOZGroupInfoList getGroupListAuthToItem(Object sessionID, String itemID,
int perm) throws OZRepositoryException
```

Gets the information of all groups with permission level of or higher than Perm for the item ID.

Parameters:

sessionID - Session ID

itemID - Item ID

perm - Permission information

getItemListAuthToGroup

```
public IOZItemInfoList getItemListAuthToGroup(Object sessionID, String groupID,  
int perm) throws OZRepositoryException
```

Gets the information of all items with permission level of or higher than Perm for the group corresponding to the designated group ID.

Parameters:

sessionID - Session ID

groupID - Group ID

perm - Permission information

IOZRepositoryAuthUserCategory

This interface provides functions to set category permissions per user.

Method Detail

getAccessTypeAuthUserCategory

```
public int getAccessTypeAuthUserCategory()
```

Gets the methods with access to the AuthUserCategory Interface as numbers.

- int ACCESS_TYPE_AUTH_USER_CATEGORY_NOT = 0x00000000
- int ACCESS_TYPE MODIFY_CATEGORY_USERAUTH = 0x00000100
- int ACCESS_TYPE_GET_USERAUTH_TO_CATEGORY = 0x00000200
- int ACCESS_TYPE_GET_USERLIST_AUTH_TO_CATEGORY = 0x00000400
- int ACCESS_TYPE_GET_ITEMLIST_AUTH_TO_USER_IN_CATEGORY= 0x00000800
- int ACCESS_TYPE_GET_CATEGORYLIST_AUTH_TO_USER_IN_CATEGORY = 0x00020000

getOZRepository

```
public IOZRepository getOZRepository() throws OZRepositoryException
```

Gets the class where the OZRepository Interface is implemented.

modifyCategoryUserAuth

```
public boolean modifyCategoryUserAuth(Object sessionID, String userID, String categoryID, int perm) throws OZRepositoryException
```

Modifies the permission information of the designated user ID for the designated category ID.

Parameters:

sessionID - Session ID

userID - User ID

categoryID - Category ID

perm - Permission information

getUserAuthToCategory

```
public int getUserAuthToCategory(Object sessionID, String userID, String categoryID) throws OZRepositoryException
```

Gets the permission information of the user ID for the designated category ID.

Parameters:

sessionID - Session ID

userID - User ID

categoryID - Category ID

getUserListAuthToCategory

```
public IOZUserInfoList getUserListAuthToCategory(Object sessionID, String categoryID, int perm) throws OZRepositoryException
```

Gets the user information of all users with permission of or higher than Perm for the designated category ID.

Parameters:

sessionID - Session ID

categoryID - Category ID

perm - Permission information

getItemListAuthToUserInCategory

```
public IOZItemInfoList getItemListAuthToUserInCategory(Object sessionID, String  
userID, String categoryID, int perm) throws OZRepositoryException
```

Gets the list of all items in the designated category for which the designated user has permission of or higher than Perm.

Parameters:

sessionID - Session ID

userID - User ID

categoryID - Category ID

perm - Permission information

getCategoryListAuthToUserInCategory

```
public IOZCategoryInfoList getCategoryListAuthToUserInCategory(Object sessionID,  
String userID, String categoryID, int perm) throws OZRepositoryException
```

Gets the list of all categories in the sub category of the designated category for which the designated user has permission of or higher than Perm.

Parameters:

sessionID - Session ID

userID - User ID

categoryID - Category ID

perm - Permission information

IOZRepositoryAuthUserItem

This interface provides functions to set item permissions per user.

Method Detail

getAccessTypeAuthUserItem

```
public int getAccessTypeAuthUserItem()
```

Gets the methods with access to the AuthUserItem Interface as numbers.

- int ACCESS_TYPE_AUTH_USER_ITEM_NOT = 0x00000000
- int ACCESS_TYPE MODIFY ITEM USERAUTH = 0x00001000
- int ACCESS_TYPE GET USERAUTH TO ITEM = 0x00002000
- int ACCESS_TYPE GET USERLIST AUTH TO ITEM = 0x00004000
- int ACCESS_TYPE GET ITEMLIST AUTH TO USER = 0x00008000

getOZRepository

```
public IOZRepository getOZRepository() throws OZRepositoryException
```

Gets the class where the OZRepository Interface is implemented.

modifyItemUserAuth

```
public boolean modifyItemUserAuth(Object sessionID, String userID, String itemID,  
int perm) throws OZRepositoryException
```

Modifies the permission information of the designated user.

Parameters:

sessionID - Session ID

userID - User ID

itemID - Item ID

perm - Permission information

getUserAuthToItem

```
public int getUserAuthToItem(Object sessionID, String userID, String itemID)  
throws OZRepositoryException
```

Gets the permission information of a user for the item.

Parameters:

sessionID - Session ID

userID - User ID

itemID - Permission information

getUserListAuthToItem

```
public IOZUserInfoList getUserListAuthToItem(Object sessionID, String itemID, int  
perm) throws OZRepositoryException
```

Gets the information of all items for which the AuthToItem is or higher than Perm.

Parameters:

sessionID - Session ID

itemID - Item ID

perm - Permission information

getItemListAuthToUser

```
public IOZItemListInfoList getItemListAuthToUser(Object sessionID, String userID, int  
perm) throws OZRepositoryException
```

Gets the information of all items of which the AuthToUser is or higher than Perm.

Parameters:

sessionID - Session ID

userID - Item ID

perm - Permission information

IOZRepositoryCategory

This interface provides functions for setting category.

Method Detail

getAccessTypeCategory

```
public int getAccessTypeCategory()
```

Gets methods accessible to the category.

- int ACCESS_CATEGORY_NOT = 0x00000000
- int ACCESS_CREATEITEM_IN_CATEGORY = 0x00000001
- int ACCESS_CREATE_CATEGORY = 0x00000002
- int ACCESS MODIFY_CATEGORY_NAME = 0x00000004
- int ACCESS_DELETE_CATEGORY = 0x00000008
- int ACCESS_UN_DELETE_CATEGORY = 0x00000010
- int ACCESS_HAS_THE_ITEM_IN_CATEGORY = 0x00000020
- int ACCESS_GET_ITEMCOUNT_IN_CATEGORY = 0x00000040
- int ACCESS_GET_ITEMLIST_IN_CATEGORY = 0x00000080
- int ACCESS_GET_CATEGORYID_OF_ITEM = 0x00000100
- int ACCESS_GET_CATEGORY_INFO = 0x00000200
- int ACCESS_GET_DELETED_ITEMLIST_IN_CATEGORY = 0x00000400
- int ACCESS_GET_SEARCH_ITEMLIST_IN_CATEGORY = 0x00000800
- int ACCESS_GET_CATEGORYLIST_IN_CATEGORY = 0x00001000
- int ACCESS_TRANSFER_ITEM = 0x00002000
- int ACCESS_TRANSFER_CATEGORY = 0x00004000

getOZRepository

```
public IOZRepository getOZRepository() throws OZRepositoryException
```

Gets the class where the OZRepository Interface is implemented.

createItemInCategory

```
public String[] createItemInCategory(Object sessionID, String[] itemNames,  
String[] descs, String[] categoryIDs, InputStream[] item_ins, String comment,  
int[] errorCode, String[] errorMsg) throws OZRepositoryException
```

Creates a new item and returns its ID.

Parameters:

sessionID - Session ID

itemName - ID of the new item

descs - Description of the new item

categoryIDs - Category ID of the new item

item_ins - Input stream of the new item

comment - Comment

errorCode - Error code

errorMsg - Error message

createCategory

```
public String[] createCategory(Object sessionID, String[] categoryName, String[]  
pCategoryIDs, String comment, int[] errorCode, String[] errorMsg) throws  
OZRepositoryException
```

Creates a new category and returns its ID.

Parameters:

sessionID - Session ID

categoryName - Name of the new category

pCategoryIDs - Parent category ID

comment - Comment

errorCode - Error code

errorMsg - Error message

modifyCategoryName

```
public String modifyCategoryName(Object sessionID, String categoryID, String  
new_CategoryName, String comment) throws OZRepositoryException
```

Modifies the name of the category corresponding to the designated category ID.

Parameters:

sessionID - Session ID

categoryID - ID of the category

new_categoryName - New category name

comment - Comment

deleteCategory

```
public boolean[] deleteCategory(Object sessionID, String[] categoryIDs, boolean[]  
isDestroys, String comment, int[] errorCode, String[] errorMsg) throws  
OZRepositoryException
```

Gets the category corresponding to the designated category ID.

Parameters:

sessionID - Session ID

categoryIDs - ID of the category to delete

isDestroys - Whether to permanently delete the category

comment - Comment

errorCode - Error code

errorMsg - Error message

unDeleteCategory

```
public boolean[] unDeleteCategory(Object sessionID, String[] categoryIDs, String  
comment, int[] errorCode, String[] errorMsg) throws OZRepositoryException
```

Rolls back a deleted category and gets the result.

Parameters:

sessionID - Session ID

categoryIDs - ID of the category to roll back

comment - Comment

errorCode - Error code

errorMsg - Error message

hasTheItemInCategory

```
public boolean hasTheItemInCategory(Object sessionID, String itemID) throws  
OZRepositoryException
```

Gets whether a specific item exists in the corresponding category.

Parameters:

sessionID - Session ID

itemID - Item ID

getItemCountInCategory

```
public int getItemCountInCategory(Object sessionID, String categoryID) throws  
OZRepositoryException
```

Gets the count of all items belonging to the designated category.

Parameters:

sessionID - Session ID

categoryID - Category ID from which to get the number of items

getItemListInCategory

```
public IOZItemInfoList getItemListInCategory(Object sessionID, String categoryID)
throws OZRepositoryException
```

Gets the information of all items belonging to the designated category.

Parameters:

sessionID - Session ID

categoryID - ID of the category from which the item information will be returned

getCategoryIdOfItem

```
public String getCategoryIDOfItem(Object sessionID, String itemID) throws
OZRepositoryException
```

Returns the ID of the category where the item exists.

Parameters:

sessionID - Session ID

itemID - ID of the item from which the category ID will be returned

getCategoryInfo

```
public IOZCategoryInfo getCategoryInfo(Object sessionID, String categoryID) throws
OZRepositoryException
```

Gets the information of the designated category.

Parameters:

sessionID - Session ID

categoryID - Category ID from which to get information

getDeletedItemListInCategory

```
public IOZItemInfoList getDeletedItemListInCategory(Object sessionID, String
categoryID) throws OZRepositoryException
```

Gets the information of deleted items in the designated category.

Parameters:

sessionID - Session ID

categoryID - Category ID from which to get information of deleted items

transferItem

```
public boolean transferItem(Object sessionID, String[] itemIDs, String target_CategoryID) throws OZRepositoryException
```

Transfers the category of the item.

Parameters:

sessionID - Session ID

itemIDs - Item ID to move its category

target_CategoryID - ID of the category to transfer

transferCategory

```
public boolean transferCategory(Object sessionID, String categoryID, String target_CategoryID) throws OZRepositoryException
```

Transfers the designated category to another category.

Parameters:

sessionID - Session ID

categoryID - ID of the source category from which to transfer

target_CategoryID - ID of the target category to which to transfer

IOZRepositoryExternalLogin

This interface provides methods to log in with a user ID created using an external API.

Method Detail

createUser

```
public String createUser(com.forcs.log4oz.OZLog log, String id, String pwd) throws  
OZRepositoryException
```

Creates a user ID using an external API.

Parameters:

log - Log

id - User ID

pwd - User password

login

```
public String login(com.forcs.log4oz.OZLog log, String id, String pwd) throws  
OZRepositoryException
```

Log in with the user ID created using an external API.

Parameters:

log - Log

id - User ID

pwd - User password

IOZRepositoryGroup

This interface provides functions for setting group.

Method Detail

getAccessTypeGroup

```
public int getAccessTypeGroup()
```

Gets the methods with access to the Group Interface as numbers.

- int ACCESS_GROUP_NOT = 0x00000000
- int ACCESS_CREATE_GROUP = 0x00000001
- int ACCESS MODIFY_GROUP_NAME = 0x00000002
- int ACCESS MODIFY_GROUP_DESC = 0x00000004
- int ACCESS_DELETE_GROUP = 0x00000008
- int ACCESS_CREATE_UESR_IN_GROUP = 0x00000010
- ACCESS_TRANSFER_USER = 0x00000020
- int ACCESS_GET_USERINLIST_IN_GROUP = 0x00000040
- int ACCESS_GET_GROUPINFO = 0x00000080
- int ACCESS_GET_SUB_GROUPINFO_LIST = 0x00000100
- int ACCESS_GET_PARENT_GROUPINFO = 0x00000200
- int ACCESS_GET_GROUPID_OF_USER = 0x00000400
- int ACCESS_ADD_GROUP_ADMIN = 0x00000800
- int ACCESS_REMOVE_GROUP_ADMIN = 0x00001000
- int ACCESS_IS_USER_GROUP_ADMIN = 0x00002000
- int ACCESS_GET_USER_ADMINLIST_IN_GROUP = 0x00004000
- int ACCESS_TRANSFER_GROUP = 0x00008000

getOZRepository

```
public IOZRepository getOZRepository() throws OZRepositoryException
```

Gets the class where the OZRepository Interface is implemented.

createGroup

```
public String createGroup(Object sessionID, String gName, String pGroupID, String desc) throws OZRepositoryException
```

Creates a new group and returns it ID.

Parameters:

sessionID - Session ID

gName - New group name to create

pGroupID - Parent group ID of the new group

desc - Description of the new group

modifyGroupName

```
public String modifyGroupName(Object sessionID, String groupID, String gName)  
throws OZRepositoryException
```

Modifies the name of the group corresponding to the designated group ID and returns the modified group ID.

Parameters:

sessionID - Session ID

groupID - ID of the group of which the name will be modified

gName - Group name

modifyGroupDesc

```
public boolean modifyGroupDesc(Object sessionID, String groupID, String desc)  
throws OZRepositoryException
```

Modifies the description of the group corresponding to the designated group ID.

Parameters:

sessionID - Session ID

groupID - ID of the group of which the description will be modified

desc - Description of the group

deleteGroup

```
public boolean deleteGroup(Object sessionID, String groupID) throws  
OZRepositoryException
```

Deletes the group corresponding to the designated group ID.

Parameters:

sessionID - Session ID

groupID - ID of the group to delete

createUserInGroup

```
public String createUserInGroup(Object sessionID, String userName, String  
password, String groupID, String desc) throws OZRepositoryException
```

Creates a new user ID in the designated group and returns the newly created user ID.

Parameters:

sessionID - Session ID

userName - User name

password - Password

groupID - Group ID

desc - User description

getUserInfoListInGroup

```
public IOZUserInfoList getUserInfoListInGroup(Object sessionID, String groupID)  
throws OZRepositoryException
```

Gets the information of all users registered in the designated group ID.

Parameters:

sessionID - Session ID

groupID - Group ID to get its user information

getGroupInfo

```
public IOZGroupInfo getGroupInfo(Object sessionID, String groupID) throws  
OZRepositoryException
```

Gets the information of the group corresponding to the designated group ID.

Parameters:

sessionID - Session ID

groupID - Group ID

getSubGroupInfoList

```
public IOZGroupInfoList getSubGroupInfoList(Object sessionID, String groupID)  
throws OZRepositoryException
```

Gets the information of the sub group of the designated group.

Parameters:

sessionID - Session ID

groupID - Group ID to get its subgroup information

getParentGroupInfo

```
public IOZGroupInfo getParentGroupInfo(Object sessionID, String groupID) throws  
OZRepositoryException
```

Gets the parent group information of the designated group.

Parameters:

sessionID - Session ID

groupID - Group ID to get its parent group information

getGroupIdOfUser

```
public IOZGroupInfo getGroupIdOfUser(Object sessionID, String userID) throws  
OZRepositoryException
```

Gets the information of the group which the designated user belong to.

Parameters:

sessionID - Session ID

userID - ID of the user from which the group information will be received

addGroupAdmin

```
public boolean addGroupAdmin(Object sessionID, String userID, String groupID)  
throws OZRepositoryException
```

Adds a Group Admin to the designated group.

Parameters:

sessionID - Session ID

userID - ID of the user to add as Group Admin will be added

groupID - ID of the group where the Group Admin will be added

removeGroupAdmin

```
public boolean removeGroupAdmin(Object sessionID, String userID, String groupID)  
throws OZRepositoryException
```

Releases a Group Admin of the designated group.

Parameters:

sessionID - Session ID

userID - ID of user from which the Group Admin authorization will be released

groupID - Group ID to remove from the user

isUserGroupAdmin

```
public boolean isUserGroupAdmin(Object sessionID, String userID, String groupID)  
throws OZRepositoryException
```

Checks that the user corresponding to the designated user ID is the administrator of the corresponding group.

Parameters:

sessionID - Session ID

userID - User ID to check if it is the group administrator

groupID - Group ID

getUserAdminListInGroup

```
public IOZUserInfoList getUserAdminListInGroup(Object sessionID, String groupID)  
throws OZRepositoryException
```

Gets the group administrator information of the designated group.

Parameters:

sessionID - Session ID

groupID - ID of group from which the group administrator information will be received

transferUser

```
public boolean transferUser(Object sessionID, String userID, String  
target_GroupID) throws OZRepositoryException
```

Transfers the group of the designated user.

Parameters:

sessionID - Session ID

userID - ID of the user that will transfer the group

target_groupID - Target group ID

transferGroup

```
public boolean transferGroup(Object sessionID, String groupID, String  
target_GroupID) throws OZRepositoryException
```

Transfers the designated group to another group.

Parameters:

sessionID - Session ID

groupId - ID of the group to which the designated group will be transferred

target_groupId - Target group ID

IOZRepositoryItem

This interface provides functions for setting item.

Method Detail

getAccessTypeItem

```
public int getAccessTypeItem()
```

Gets methods accessible to the item.

- int ACCESS_ITEM_NOT = 0x00000000
- int ACCESS_CREATE_ITEM = 0x00000001
- int ACCESS MODIFY_ITEMNAME = 0x00000002
- int ACCESS_DELETE_ITEM = 0x00000004
- int ACCESS_UN_DELETE_ITEM = 0x00000008
- int ACCESS_MODIFY_ITEM_DESC = 0x00000010
- int ACCESS_HAS_THE_ITEM = 0x00000020
- int ACCESS_GET_ITEM_INFO = 0x00000040
- int ACCESS_GET_ITEMS_UNCONDITION = 0x00000080
- int ACCESS_GET_ITEMS = 0x00000100
- int ACCESS_CHECKOUT_ITEM = 0x00000200
- int ACCESS_CHECKIN_ITEM = 0x00000400
- int ACCESS_UNDO_CHECKOUT_ITEM = 0x00000800
- int ACCESS_IS_CHECKOUT_USER = 0x00001000

getOZRepository

```
public IOZRepository getOZRepository() throws OZRepositoryException
```

Gets the class where the OZRepository Interface is implemented.

createItem

```
public String[] createItem(Object sessionID, String[] itemNames, String[] descs,  
InputStream[] item_ins, String comment, int[] errorCode, String[] errorMsg) throws  
OZRepositoryException
```

Creates an item and returns its ID.

Parameters:

sessionID - Session ID

itemNames - Item name

descs - Item description

item_ins - Item stream

comment - Comment

errCodes - Error code

errMsgs - Error message

modifyItemName

```
public String modifyItemName(Object sessionID, String itemID, String new_itemName,  
String comment) throws OZRepositoryException
```

Modifies the name of the item corresponding to the item ID and returns the modified item ID.

Parameters:

sessionID - Session ID

itemID - Item ID to change its name

new_itemName - New item name

comment - Comment

deleteItem

```
public boolean[] deleteItem(Object sessionID, String[] itemIDs, boolean[]  
isDestroys, String comment, int[] errorCode, String[] errorMsg) throws  
Deletes the item from the Repository and gets the result.
```

Parameters:

sessionID - Session ID

itemIDs - ID of the item to delete

isDestroys - Whether to permanently delete

comment - Comment

errCodes - Error code

errMsgs - Error message

unDeleteItem

```
public boolean[] unDeleteItem(Object sessionID, String[] itemIDs, String comment,  
int[] errorCode, String[] errorMsg) throws OZRepositoryException
```

Restores the deleted item.

Parameters:

sessionID - Session ID

itemIDs - ID of the item to restore

comment - Comment

errCodes - Error code

errMsgs - Error message

modifyItemDesc

```
public boolean modifyItemDesc(Object sessionID, String itemID, String desc) throws  
OZRepositoryException
```

Modifies the description of the item corresponding to the item ID.

Parameters:

sessionID - Session ID

itemID - ID of item to modify

desc - Description to modify

getItemInfo

```
public IOZItemInfo getItemInfo(Object sessionID, String itemID) throws  
OZRepositoryException
```

Gets the information of the item corresponding to the item ID.

Parameters:

sessionID - Session ID

itemID - ID of item from which to get information

hasTheItem

```
public boolean hasTheItem(Object sessionID, String itemID) throws  
OZRepositoryException
```

Gets whether the item exists.

Parameters:

sessionID - Session ID

itemID - Item ID

getItems

```
public InputStream[] getItems(Object sessionID, String itemIDs[], int[] errorCode,  
String[] errorMsg) throws OZRepositoryException
```

Gets the item and the item stream.

Parameters:

sessionID - Session ID

itemIDs - Item ID to get

errCodes - Error code

errMsgs - Error message

getItems

```
public InputStream[] getItems(Object sessionID, String itemIDs[], long[]
modifiedTimes, int[] errorCode, String[] errorMsg) throws OZRepositoryException
```

Gets the item and the item stream.

Parameters:

sessionID - Session ID

itemIDs - Item ID to get

modifiedTimes - Time the item was last modified

errCodes - Error code

errMsgs - Error message

checkOutItem

```
public InputStream[] checkOutItem(Object sessionID, String itemIDs[], String[]
localCheckOutFolders, long[] localFileTimes, String[] checkOutCmts, int[]
errorCode, String[] errorMsg) throws OZRepositoryException
```

Check out the item with the designated user ID to a folder, and gets the item stream if the local file time is smaller than the item time.

Parameters:

sessionID - Session ID

itemIDs - ID of the item to check out

localCheckOutFolders - Name of the local folder to check out

localFileTimes - Local file time of the item to check out

checkOutCmts - Comment

errCodes - Error code

errMsgs - Error message

checkInItem

```
public boolean[] checkInItem(Object sessionID, String[] itemIDs, InputStream[]  
item_ins, String comment, boolean[] keepCheckOut, int[] errorCode, String[]  
errorMsg) throws OZRepositoryException
```

Check in the item with the designated user ID.

Parameters:

sessionID - Session ID

itemIDs - ID of the item to check in

item_ins - Item stream to check in

comment - Comment

keepCheckOut - Whether to keep the check out status

errCodes - Error code

errMsgs - Error message

undoCheckOutItem

```
public InputStream[] undoCheckOutItem(Object sessionID, String[] itemIDs,  
boolean[] isReplaces, int[] errorCode, String[] errorMsg) throws  
OZRepositoryException
```

Undo check out the item.

Parameters:

sessionID - Session ID

itemIDs - ID of the item to undo check out

isReplaces - Whether to replace the local file

errCodes - Error code

errMsgs - Error message

isCheckOutUser

```
public boolean[] isCheckOutUser(Object sessionID, String[] itemIDs) throws  
OZRepositoryException
```

Checks whether the designated user checked out the item.

Parameters:

sessionID - Session ID

itemIDs - Item ID

IOZRepositoryItemHistory

This interface provides information on the item's change history.

Method Detail

getAccessTypeItemHistory

```
public int getAccessTypeItemHistory()
```

Gets the methods with access to the Item History Interface as numbers.

- int ACCESS_ITEM_HISTORY_NOT = 0x00000000
- int ACCESS_ROLLBACK_ITEM = 0x00000001
- int ACCESS_GET_ITEM_BY_VERSION = 0x00000002
- int ACCESS_GET_HISTORY_ITEMLIST = 0x00000004
- int ACCESS_GET_DELETE_HISTORY_ITEMLIST = 0x00000008
- int ACCESS_REMOVE_HISTORY_ITEM = 0x00000010

getOZRepository

```
public IOZRepository getOZRepository() throws OZRepositoryException
```

Gets the class where the OZRepository Interface is implemented.

rollBackItem

```
public boolean rollBackItem(Object sessionID, String itemID, int itemVersion,  
String comment) throws OZRepositoryException
```

Rolls back a item to a designated version.

Parameters:

sessionID - Session ID

itemID - ID of the item to roll back

itemVersion - Version to roll back

comment - Comment

getItemByVersion

```
public InputStream getItemByVersion(Object sessionID, String itemID, int version)  
throws OZRepositoryException
```

Gets the item of the designated version among the items corresponding to the item ID.

Parameters:

sessionID - Session ID

itemID - ID of the item to get

version - The version of the item to get

getHistoryItemList

```
public IOZHistoryInfoList getHistoryItemList(Object sessionID, String itemID)  
throws OZRepositoryException
```

Gets the history information of the item.

Parameters:

sessionID - Session ID

itemIDs - Item ID from which to get history

getDeleteHistoryItemInfo

```
public IOZHistoryInfoList getDeleteHistoryItemInfo(Object sessionID, String itemID) throws OZRepositoryException
```

Gets the deleted history list.

Parameters:

sessionID - Session ID

itemIDs - Item ID from which to get deleted history

removeHistoryItem

```
public boolean removeHistoryItem(Object sessionID, String itemID, int itemVersion) throws OZRepositoryException
```

Deletes the item's specific version of history.

Parameters:

sessionID - Session ID

itemID - Item ID to delete its history

item version - History version to be deleted

IOZRepositoryMultiLoginUser

This interface provides multiple login related methods.

Method Detail

getAccessTypeMultiLoginUser

```
public int getAccessTypeMultiLoginUser()
```

Gets the methods with access to the MultiLogin Interface as numbers.

- int ACCESS_MULTILOGINUSER_NOT = 0x00000000
- int ACCESS_DISABLE_USER_LOGIN = 0x00000001
- int ACCESS_ENABLE_USER_LOGIN = 0x00000002
- int ACCESS_IS_LOGIN_ENABLED = 0x00000004

getOZRepository

```
public I0ZRepository getOZRepository() throws OZRepositoryException
```

Gets the class where the OZRepository Interface is implemented.

disableUserLogin

```
public void disableUserLogin(Object sessionID) throws OZRepositoryException
```

Set to disable the Userlogin with the designated session ID.

Parameters:

sessionID - The designated session ID

enableUserLogin

```
public void enableUserLogin(Object sessionID) throws OZRepositoryException
```

Set to enable the UserLogin with the designated session ID.

Parameters:

sessionID - The designated session ID

isLoginEnabled

```
public boolean isLoginEnabled(Object sessionID) throws OZRepositoryException
```

Gets whether the login of the user with the designated session ID is enabled.

Parameters:

sessionID - Session ID to check if it can be logged in

IOZRepositoryUser

This interface provides functions for setting user.

Method Detail

getAccessTypeUser

```
public int getAccessTypeUser()
```

Gets the methods with access to the User Interface as numbers.

- int ACCESS_USER_NOT = 0x00000000
- int ACCESS_CREATE_USER = 0x00000001
- int ACCESS MODIFY_USER_NAME = 0x00000002
- int ACCESS MODIFY_USER_PASSWORD = 0x00000004
- int ACCESS MODIFY_USER_DESC = 0x00000008
- int ACCESS_DELETE_USER = 0x00000010
- int ACCESS_GET_USERINFO = 0x00000020
- int ACCESS_CHECK_USER_PASSWORD = 0x00000040
- int ACCESS_GET_USERINFO_LIST = 0x00000080
- int ACCESS_GET_IS_CHECK_ADMIN = 0x00000100

getOZRepository

```
public IOZRepository getOZRepository() throws OZRepositoryException
```

Gets the class where the OZRepository Interface is implemented.

createUser

```
public String createUser(Object sessionID, String userName, String password,  
String desc) throws OZRepositoryException
```

Creates a new user and returns its ID.

Parameters:

sessionID - Session ID

userName - User name

password - Password

desc - User description

modifyUserName

```
public String modifyUserName(Object sessionID, String userID, String userName)  
throws OZRepositoryException
```

Modifies the name of the user corresponding to the user ID and returns the ID of which the user name was modified.

Parameters:

sessionID - Session ID

userID - User ID

userName - User name to modify

modifyUserPassword

```
public boolean modifyUserPassword(Object sessionID, String userID, String  
old_password, String new_password) throws OZRepositoryException
```

Modifies the password of the designated user.

Parameters:

sessionID - Session ID

userID - User ID to change its password

old_password - Old password

new_password - New password

modifyUserDesc

```
public boolean modifyUserDesc(Object sessionID, String userID, String desc) throws  
OZRepositoryException
```

Modifies the description of the user corresponding to the designated user ID.

Parameters:

sessionID - Session ID

userID - User ID to change its user description

desc - Description of the user to modify

deleteUser

```
public boolean deleteUser(Object sessionID, String userID) throws  
OZRepositoryException
```

Deletes the information of all users corresponding to the designated user ID.

Parameters:

sessionID - Session ID

userID - ID of the user to delete

getUserInfo

```
public IOZUserInfo getUserInfo(Object sessionID, String userID) throws  
OZRepositoryException
```

Gets the user information of the user corresponding to the user ID.

Parameters:

sessionID - Session ID

userID - User ID to get user information

checkUserPassword

```
public boolean checkUserPassword(Object sessionID, String userID, String password)  
throws OZRepositoryException
```

Checks whether the user password is correct.

Parameters:

sessionID - Session ID

userID - User ID to check its password

password - Password to check

getUserInfoList

```
public I0ZUserInfoList getUserInfoList(Object sessionID) throws  
OZRepositoryException
```

Gets the user information of all users.

Parameters:

sessionID - Session ID

isCheckAdmin

```
public boolean isCheckAdmin(Object sessionID, String userID) throws  
OZRepositoryException
```

Checks that the user corresponding to the designated user ID is the administrator of the Repository.

Parameters:

sessionID - Session ID

userID - User ID to check if it is the administrator

addAdmin

```
public boolean addAdmin(Object sID, String uID, String client_ip) throws  
OZRepositoryException
```

Adds the administrator.

Parameters:

sID - Session ID

uID - User ID for adding the administrator permission

client_ip - Client IP

removeAdmin

```
public boolean removeAdmin(Object sID, String uID, String client_ip) throws  
OZRepositoryException
```

Cancel the administrator.

Parameters:

sID - Session ID

uID - User ID for deleting the administrator permission

client_ip - Client IP

modifyAllowip

```
public boolean modifyAllowip(Object sID, String uID, String allowip, String  
client_ip) throws OZRepositoryException
```

Changes the allowed IP of users.

Parameters:

sID - Session ID

uID - User ID for changing the allowed IP

allowip - Allowed IP

client_ip - Client IP

isExternalLogin

```
public boolean isExternalLogin() throws OZRepositoryException
```

Gets whether to use the login function connected to external API.

IOZUserInfo

This interface provides group information.

Method Detail

getUserID

```
public String getUserID()
```

Gets the ID of the user.

getPassword

```
public String getPassword()
```

Gets the password of the user.

getUserName

```
public String getUserName()
```

Gets the name of the user.

getGroupID

```
public String getGroupID()
```

Gets the ID of the group where the user belongs.

getGroupName

```
public String getGroupName()
```

Gets the name of the group where the user belongs.

getDesc

```
public String getDesc()
```

Gets the description of the user.

getUpdateUserTime

```
public long getUpdateUserTime()
```

Gets the update time of the user.

isLoggedIn

```
public boolean isLoggedIn()
```

Gets whether the user is logged in.

getSessionID

```
public String getSessionID()
```

Gets the session ID created when the user logs in.

isLoginEnabled

```
public boolean isLoginEnabled()
```

Gets whether login of the user is enabled.

getLoginIP

```
public String getLoginIP()
```

Gets the IP of the PC where the user logged in.

getLastLoginTime

```
public long getLastLoginTime()
```

Gets the last login time of the user.

getPerm

```
public int getPerm()
```

Gets the permission information of the user.

IOZUserInfoList

This interface provides group information.

Method Detail

getGroupInfo

```
public I0ZUserInfoList getUserInfo(int index)
```

Gets the user information corresponding tot the index.

Parameters:

index-Index

getSize

```
public String getSize()
```

Gets the number of user lists.

getGroupInfos

```
public I0ZUserInfo[] getUserInfos()
```

Gets the user information as array.

User Defined Log API

The UDL(User Defined Log) is an interface to allow users to record databases or information related to the report in the user-defined type in a file or DB when referring data through the OZ server.

User Defined Log API provides [classes](#) and [interfaces](#).

Class

User Defined Log API provides the following class:

- [OZUserDefinedLogger](#)

OZUserDefinedLogger

This class provides methods to implement User Defined Log.

Method Detail

init

```
public void init(oz.framework.db.OZConnection ozconn, com.forcs.log4oz.OZLog cat)  
throws OZUserDefinedLogException
```

Defines the object where the user-defined log will be saved when executing the OZ Server.

Parameters:

ozconn - The DB connection data and the log type to be used when saving user-defined log in the DB

cat - The file information and log type to be used when saving user-defined log in a file

traceLog

```
public void traceLog(oz.udl.OZUserDefinedLogTarget target,  
oz.framework.db.OZConnection ozconn, com.forcs.log4oz.OZLog cat) throws  
OZUserDefinedLogException
```

Defines the target content to save in the log.

Parameters:

target - User-defined log target

ozconn - The DB connection data and the log type to be used when saving user-defined log in the DB.

cat - The file information and log type to be used when saving user-defined log in the file.

getOZDBAlias

```
public String getOZDBAlias()
```

Gets the alias of the connection pool defined in OZ Server.

Interface

User Defined Log API provides the following Interfaces:

- [OZClearQuestLogTarget](#)
- [OZDMLogTarget](#)
- [OZEndTarget](#)
- [OZExecuteTarget](#)
- [OZFileStoreLogTarget](#)
- [OZHTTPStoreLogTarget](#)
- [OZMonitorLogTarget](#)
- [OZProcedureLogTarget](#)
- [OZRDBLogTarget](#)
- [OZRequestTarget](#)
- [OZSAPLogTarget](#)
- [OZSchedulerUDLTarget](#)
- [OZSchedulerUserDefinedLogger](#)
- [OZSOAPLogTarget](#)
- [OZTaskInfoTarget](#)
- [OZUDSLogTarget](#)
- [OZUserDefinedLogTarget](#)
- [OZXMLLogTarget](#)

OZClearQuestLogTarget

This interface provides ClearQuest store related information.

Method Detail

getExecuteType

```
public String getExecuteType()
```

Gets the Clear Quest execution type requested by the client.

getQuerySubType

```
public String getQuerySubType()
```

Gets the QuerySub type requested by the client.

getQuery

```
public String getQuery()
```

Gets the query statement requested by the client.

getQueryExecuteTime

```
public long getQueryExecuteTime()
```

Gets the time taken for the query statement requested by the client to execute.(Unit:
Milliseconds)

getQueryState

```
public String getQueryState()
```

Gets whether the query statement requested by the client successfully executed.

OZDMLogTarget

This interface provides information about the data module.

Method Detail

getODIName

```
public String getODIName()
```

Gets the name of the ODI that executed the query statement.

getDataSetName

```
public String getDataSetName()
```

Gets the name of the data set that executed the query statement.

getDataStoreName

```
public String getDataStoreName()
```

Gets the name of the data store that executed the query statement.

getOZParam

```
public HashMap getOZParam()
```

Gets the OZ user-defined parameters in the form of HashMap.

OZEndTarget

This interface is the interface that is called when the task ends.

Method Detail

getExecuteTime

```
long getExecuteTime()
```

Gets the execution time in milliseconds.

getExportedFileName

`String[] getExportedFileName()`

Gets the name of the exported file.

isSuccessfulString

`String isSuccessfulString()`

Gets the message indicating whether successful or not.

getErrorMsg

`String getErrorMsg()`

Gets the error message.

OZExecuteTarget

This interface is the interface that is called when executing a task.

Method Detail

getReportFiles

`String[] getReportFiles()`

Gets the report name with category path.

getSchedulingType

`String getSchedulingType()`

Gets the scheduling type.

Return "Immediately", "Once at specific time" or "Periodically".

getViewParamString

`String getViewerParamString()`

Gets viewer parameters.

OZFileStoreLogTarget

This interface provides file store related information.

Method Detail

getFilePath

`public String getFilePath()`

Gets the file path requested by the client.

getFileAccessTime

`public long getFileAccessTime()`

Gets the time taken for the file requested by the client to be received.(Unit: Milliseconds)

getFileAccessState

`public String getFileAccessState()`

Gets whether the file requested by the client successfully executed.

OZHTTPStoreLogTarget

This interface provides HTTP store related information.

Method Detail

getURL

```
public String getURL()
```

Gets the URL requested by the client.

getHTTPAccessTime

```
public long getHTTPAccessTime()
```

Gets the time taken for the files in the HTTP path requested by the client to be received.(Unit: Milliseconds)

getHTTPAccessState

```
public String getHTTPAccessState()
```

Gets whether the file of the HTTP path requested by the client successfully executed.

If successful return "true", else return "false:error message".

OZMonitorLogTarget

This interface provides monitor log related information.

Method Detail

getMark

```
public String getMark()
```

Gets the signs of the service start and end.

getThreadName

```
public String getThreadName()
```

Gets the service thread name.

getServiceTime

```
public long getServiceTime()
```

Gets the service start time and end time.(Unit: Milliseconds)

getFreeMemory

```
public long getFreeMemory()
```

Gets the free memory.(Unit: byte)

getTotalMemory

```
public long getTotalMemory()
```

Gets the total JVM memory size.(Unit: byte)

getServiceCode

```
public integer getServiceCode()
```

Gets the service code.

getServiceStatus

```
public integer getServiceStatus()
```

Gets the service status code.

getServiceParameter

```
public String getServiceParameter()
```

Gets the parameter value that is transmitted when requesting a service in the client.

getDBSessionID

```
public String getDBSessionID()
```

Gets the DBMS session ID.

getExecuteTime

```
public String getExecuteTime()
```

Gets the time taken to process a service.(Unit: Milliseconds)

getErrorCode

```
public String getErrorCode()
```

Gets the error code when an error occurs.

getErrorMsg

```
public String getErrorMsg()
```

Gets the error message when an error occurs

getErrorStackTrace

```
public String getErrorStackTrace()
```

Gets the Stack Trace when an error occurs.

getDBConns

```
public String getDBConns()
```

Gets the number of DB connections by Alias.

OZProcedureLogTarget

This interface provides information about procedure data.

Method Detail

getDBAlias

```
public String getDBAlias()
```

Gets the DB alias of the connection pool used by the executed ODI.

getProcedureName

public String getProcedureName()

Gets the name of the procedure requested by the client.

getParameterList

public ArrayList getParameterList()

Gets the procedure parameter requested by the client as array.

getProcedureExecuteTime

public long getProcedureExecuteTime()

Gets the time take for the procedure requested by the client to execute.(Unit: Milliseconds)

getProcedureState

public String getProcedureState()

Gets whether the procedure requested by the client has successfully executed.

OZRDBLogTarget

This interface provides information about the RDB store.

Method Detail

getDBAlias

public String getDBAlias()

Gets the DB alias of the connection pool used by the executed ODI.

getDBExecuteQuery

public String getDBExecuteQuery()

Gets the query statement requested by the client.

getPreparedQueryValue

```
public Vector getPreparedQueryValue()
```

The used value for the complied query statement.

getExecuteQueryState

```
public String getExecuteQueryState()
```

Gets whether the query statement requested by the client has successfully been executed.

getQueryExecuteTime

```
public long getQueryExecuteTime()
```

Gets the time taken for the query statement requested by the client to be executed.(Unit: Milliseconds)

isPrepared

```
public boolean isPrepared()
```

Gets whether a compiled query statement was used.

getCRUDFlagName

```
public String getCRUDFlagName()
```

Gets the type of the query statement executed and requested by the client.(select, insert, update, delete)

OZRequestTarget

This interface is the interface that is called upon task request.

Method Detail

getConfigType

```
String getConfigType()
```

Gets the configuration type.

OZSAPLogTarget

This interface provides SAP store related information.

Method Detail

getFunctionName

```
public String getFunctionName()
```

Gets the name of the function requested by the client.

getFunctionType

```
public String getFunctionType()
```

Gets the function type requested by the client.

getInputParameters

```
public HashMap getInputParameters()
```

Gets the input parameter requested from the client in the form of HashMap.

getResultSetTypes

```
public String getResultSetTypes()
```

Gets the type of the ResultSet requested by the client.

getFunctionExecuteTime

```
public long getFunctionExecuteTime()
```

Gets the time taken for the function requested by the client to execute.(Unit: Milliseconds)

getFunctionState

```
public String getFunctionState()
```

Gets whether the function requested by the client successfully executed.

OZSchedulerUDLTarget

This interface provides a method to get basic information of UDL used by OZ Scheduler.

Method Detail

getConcurrentCount

```
int getConcurrentCount()
```

Gets the number of running tasks.

getWaitQueueCount

```
int getWaitQueueCount()
```

Gets the number of waiting tasks.

getTaskID

```
String getTaskID()
```

Gets the task ID.

getMarkType

```
String getMarkType()
```

Gets the execution type."REQ", "EXE", "END")

getFreeMemory

`long getFreeMemory()`

Gets free memory size in bytes

getTotalMemory

`long getTotalMemory()`

Gets total memory size in bytes

getServiceTime

`long getServiceTime()`

Gets the number of milliseconds elapsed since midnight(00:00:00), January 1, 1970.

getExportMap

`SortProperties getExportMap()`

Gets the execution option of the HTML5SVG viewer.

getSVGType

`String getSVGType()`

Gets the job type that the HTML5SVG viewer requests to the scheduler.(NewFrame, GetPage, CheckStatus, RequestSchedulerExport, RequestSchedulerPrint, GetMHTPage, GetPDFPage, ReqeustExport, RequestPrint)

OZSchedulerUserDefinedLogger

This interface provides basic methods to execute User Defined Log function in OZ Scheduler.

Method Detail

init

`void init(org.apache.log4j.Category cat) throws OZSchedulerUDLEException`

Starts the OZ Scheduler.

Parameters:

cat - Message to log when starting the OZ Scheduler

release

```
void release(org.apache.log4j.Category cat) throws OZSchedulerUDLEException
```

Stop the OZ Scheduler.

Parameters:

cat - Message to log when stopping the OZ Scheduler

traceLog

```
void traceLog(OZSchedulerUDLTarget target, org.apache.log4j.Category cat) throws  
OZSchedulerUDLEException
```

Write the log

Parameters:

target - Target user defined log

cat - Message to log

OZSOAPLogTarget

This interface provides SOAP store related information.

Method Detail

getServiceName

```
public String getServiceName()
```

Gets the service name requested by the client.

getPort

```
public String getPort()
```

Gets the service port requested by the client.

getOperation

```
public String getOperation()
```

Gets the operation of the service requested by the client.

getEndPoint

```
public String getEndPoint()
```

Gets the EndPoint that called the service requested by the client.

getRequestXML

```
public String getRequestXML()
```

Gets the XML content requested by the client.

getServiceExecuteTime

```
public long getServiceExecuteTime()
```

Gets the time taken for the service requested by the client to execute.(Unit: Milliseconds)

getServiceState

```
public String getServiceState()
```

Gets whether the SOAP service requested by the client successfully executed.

OZTaskInfoTarget

This interface is the interface that is called when saving task information.

Method Detail

getTaskGroupName

`String getTaskGroupName()`

Gets the task group name with full path.

getTaskName

`String getTaskName()`

Gets the task name.

getTaskProperty

`HashMap getTaskProperty()`

Gets the task information in the form of `HashMap`.

OZUDSLogTarget

This interface provides information about the user data store.

Method Detail

getExecuteCommand

`public String getExecuteCommand()`

Gets the command requested by the client.

getCRUDFlagName

`public String getCRUDFlagName()`

Gets the type of the query statement requested and executed by the client.

getCommandExecuteTime

```
public long getCommandExecuteTime()
```

Gets the time taken for the command requested by the client to execute.(Unit: Milliseconds)

getCommandState

```
public String getCommandState()
```

Gets whether the command requested by the client successfully executed.

OZUserDefinedLogTarget

This interface provides basic information of OZ UDL.

Method Detail

getIPAddress

```
public String getIPAddress()
```

Gets the OZ Server IP address requested by the client.

getHttpRequest

```
public HttpServletRequest getHttpRequest()
```

Gets the requested HttpRequest when the OZ Server type is Servlet.

getHttpServlet

```
public HttpServlet getHttpServlet()
```

Gets the requested HttpServlet when the OZ Server type is Servlet.

getUserID

```
public String getUserID()
```

Gets the OZ Server user ID that executed the query statement in the client.

OZXMLLogTarget

This interface provides XML store related information.

Method Detail

getURL

```
public String getURL()
```

Gets the file path or URL requested by the client.

getURLAccessTime

```
public long getURLAccessTime()
```

Gets the time taken for the file of the URL requested by the client to be received.(Unit:
Milliseconds)

getURLAccessState

```
public String getURLAccessState()
```

Gets whether the file of the URL requested by the client successfully executed.

User Defined Resource API

When create a resource field of a report, UDR(User Defined Resource) enables you to set the resource field values using resource information in database without setting the each resource field values using OZ Report Designer.

User Defined Resource API provides [classes](#) and [interfaces](#).

Class

User Defined Resource API provides the following classes:

- [OZUDRInfo](#)
- [OZUDRResult](#)
- [Resource](#)

OZUDRInfo

This class provides methods to get UDR information.

Constructor Detail

OZUDRInfo

```
public OZUDRInfo(String category, String reportname, String[] language, Resource[] resource)
```

Gets the UDR information.

Parameters:

category - Category name

reportname - Report name

language - Language list

resource - Resource information object

Method Detail

getCategory

```
public String getCategory()
```

Gets the category name.

setCategory

```
public void setCategory(String category)
```

Sets the category name.

Parameters:

category - Category name

getReportname

```
public String getReportname()
```

Gets the report name.

setReportname

```
public void setReportname(String reportname)
```

Sets the report name.

Parameters:

reportname - Report name

getLanguage

```
public String[] getLanguage()
```

Gets the language list.

setLanguage

```
public void setLanguage(String[] language)
```

Sets the language list.

Parameters:

language - Language list

getResource

`public Resource[] getResource()`

Gets the resource.

setResource

`public void setResource(Resource[] resource)`

Sets the resource.

Parameters:

resource - Resource information

OZUDRResult

This class provides methods to get the UDR execution result.

Constructor Detail

OZUDRResult

`public OZUDRResult(boolean isUpdate, OZUDRInfo info)`

Gets the UDR execution result.

Parameters:

isUpdate - Whether to update or not

info - An object that has new user defined resource information

Method Detail

isUpdate

```
public boolean isUpdate()
```

Whether to update or not.

getUDRInfo

```
public OZUDRInfo getUDRInfo()
```

Gets the object that has new user defined resource information.

Resource

This class provides methods to get resource information.

Constructor Detail

Resource

```
public Resource(String name, String desc, String[] value)
```

Gets resource information

Parameters:

name - Resource field name

desc - Resource description

value - Resource value for each language

Method Detail

getName

```
public String getName()
```

Gets the resource field name.

setName

```
public void setName(String name)
```

Sets the resource field name.

Parameters:

name - Resource field name

getDesc

```
public String getDesc()
```

Gets the resource description.

setDesc

```
public void setDesc(String desc)
```

Sets the resource description.

Parameters:

desc - Resource description

getValue

```
public String[] getValue()
```

Gets the resource values for each language.

setValue

```
public void setValue(String[] value)
```

Sets the resource values for each language.

Parameters:

value - Resource values for each language

getDescHash

`public HashMap getDescHash()`

Gets the object with key, value in the hash for parsing resource description.

Interface

User Defined Resource API provides the following interface:

- [IOZUserDefinedResource](#)

IOZUserDefinedResource

This interface provides resource synchronization functions.

Method Detail

syncResource

```
public OZUDRResult syncResource(OZConnection ozconn, OZLog cat, String props,  
OZUDRInfo info) throws Exception
```

Synchronize resource.

Parameters:

ozconn - DB Connection

cat - Category to be logged in server log

props - User defined Property

info - Resource information object in the report

User Component API

The methods available when developing custom components are as follows:

Method Detail

GetNewInstance

`DWORD GetNewInstance()`

User component object creation function.

GetCopyInstance

`DWORD GetCopyInstance(DWORD src)`

User component object copy function.

Parameters:

`src` - User component object handle

DeleteInstance

`void DeleteInstance(DWORD src)`

User component object deletion function.

Parameters:

`src` - User component object handle

getAttrListLength

`int getAttrListLength(DWORD src)`

A function that returns the number of properties used by user components.

Parameters:

src - User component object handle

getAttrList

BOOL getAttrList(DWORD src, WCHAR** attrs, const int length)

Puts property names used in the user component and then returns them to the attrs.

Parameters:

src - User component object handle

attrs - Pointer of the buffer where the properties are returned.

length - Number of properties generated by calling the getAttrListLength.

getAttrLength

int getAttrLength(DWORD src, const WCHAR* name)

A function that returns the length of the input property.

Parameters:

src - User component object handle

name - Property name

getAttr

BOOL getAttr(DWORD src, const WCHAR* name, WCHAR* value, const int value_length)

Returns the properties corresponding to the name to the value.

Parameters:

src - User component object handle

name - Property name

value - Place where the properties are saved

value_length - Value generated by calling the getAttrLength

setAttr

```
BOOL setAttr(DWORD src, const WCHAR* name, const WCHAR* value)
```

A function that modifies the value of the input property.

Parameters:

src - User component object handle

name - Property name

value - Value of the property to modify

readMe

```
void readMe(DWORD src, const char* pData, const int length)
```

A function that read user component properties from the serialized binary data.

Parameters:

src - User component object handle

pData - Binary data storage for reading properties

length - Length of the data to read

writeMeExLength

```
int writeMeExLength(DWORD src)
```

A function that returns the length of binary data that serializes the properties of user components.

Parameters:

src - User component object handle

writeMeEx

```
void writeMeEx(DWORD src, char* pData, const int length)
```

A function that serializes the properties of user components.

Parameters:

src - User component object handle

pData - Pointer of binary data

length - Length of binary data

isHiddenAttr

```
BOOL isHiddenAttr(DWORD src, LPCWSTR attrname)
```

A function that sets whether to display the property in the properties window.

Parameters:

src - User component object handle

attrname - Property name

getAttrType

```
int getAttrType(DWORD src, LPCWSTR attrname)
```

A function that add property to match the property type.

Parameters:

src - User component object handle

attrname - Property name

getEnumSize

```
int getEnumSize(DWORD src, LPCWSTR attrname)
```

A function that get the number of Enum, when the return value of the getAttrType function is Enum.

Parameters:

src - User component object handle

attrname - Property name

getEnumSize

```
BOOL getEnumValues(DWORD src, LPCWSTR attrname, WCHAR** enumNames, const int length)
```

A function that get the values of Enum, when the return value of the getAttrType function is Enum.

Parameters:

src - User component object handle

attrname - Property name

enumNames - Array of Enum values to be stored

length - Number of Enum

paint

```
void paint(DWORD src, HDC hDC, LPCWSTR data, const float x, const float y, const float w, const float h, const float scale)
```

A function called when drawing on the screen.

Parameters:

src - User component object handle

hDC - Handle of the screen(refer to the WIN32 API)

data - Caption of the label

x - X-coordinate of the component

y - Y-coordinate of the component

w - Width of the component

h - Height of the component

scale - Scale

paintEX

```
void paintEX(DWORD src, HDC hDC, LPCWSTR data, const float x, const float y, const  
float w, const float h, const float scale, const float x_offset, const float  
y_offset)
```

A function called when drawing on the screen and that delivers the offset value where the actual image will be saved.

Parameters:

src - User component object handle

hDC - Handle of the screen(refer to the WIN32 API)

data - Caption of the label

x - X-coordinate of the component

y - Y-coordinate of the component

w - Width of the component

h - Height of the component

scale - Scale

x_offset - x offset where the actual image will be saved

y_offset - y offset where the actual image will be saved

print

```
void print(DWORD src, HDC hDC, LPCWSTR data, const float x, const float y, const  
float w, const float h, const float scale, const float x_offset, const float  
y_offset)
```

A function that is called when printing.

Parameters:

src - User component object handle

hDC - Handle of the screen(refer to the WIN32 API)

data - Caption of the label

x - X-coordinate of the component

y - Y-coordinate of the component

w - Width of the component

h - Height of the component

scale - Scale

x_offset - x offset where the actual image will be saved

y_offset - y offset where the actual image will be saved

getAutosize

```
void getAutosize(DWORD src, HDC hDC, LPCWSTR data, float * w, float * h)
```

A function to set the horizontal and vertical size of the user component when using the AutoSize function.

Parameters:

src - User component object handle

hDC - Handle of the screen(refer to the WIN32 API)

data - Caption of the label

w - pointer of the original width of the component

h - pointer of the original height of the component

setWorkPath

```
void setWorkPath(DWORD src, const WCHAR* work_path)
```

A function that set the path of working folder.

Parameters:

src - User component object handle

work_path - Path of working folder

setDLLPath

```
void setDLLPath(DWORD src, const WCHAR* dll_path)
```

A function that the path of DLL file.

Parameters:

src - User component object handle

dll_path - Path of DLL file

setExtraParam

```
void setExtraParam(DWORD src, const WCHAR* extra_param)
```

A function that apply parameter.

Parameters:

src - User component object handle

extra_param - Parameter

getTooltipLength

```
int getTooltipLength(DWORD src, const float x, const float y)
```

A function that get the length of a tooltip.

Parameters:

src - User component object handle

x - X-coordinate of tooltip

y - Y-coordinate of tooltip

getTooltip

```
BOOL getTooltip(DWORD src, const float x, const float y, WCHAR* tooltip, const int tooltip_length)
```

A function that get whether tooltip is displayed.

Parameters:

src - User component object handle

x - X-coordinate of tooltip

y - Y-coordinate of tooltip

tooltip - Text of tooltip

tooltip_length - Length of tooltip

splitBindEx

```
BSTR splitBindEx(DWORD src, HDC hDC, LPCWSTR data, const float firstHeiht, int  
firstcount, const float nextHeiht, const float desiredWidth)
```

A function of applying the "User Component" in the Drawing Type of Note.

Parameters:

src - User component object handle

hDC - Handle of the screen(refer to the WIN32 API)

data - Data

firstHeiht - First height

firstcount - First count

nextHeiht - Next height

desiredWidth - Desired Width