

Report and Folder Storage/Management

By default, Exago stores the reports in a file system folder. The location of this folder is specified in the 'Report Path' property set in the Administration Console. Alternatively, report, template and folder storage, and retrieval can be handled by building a Web Service or .NET Assembly. This would allow for reports, folders and templates to be stored in a database. To do this, specify the Web Service or .NET Assembly in the Report Path of **Main Settings**. The Web Service or .NET Assembly should contain all of the methods specified in **List of Methods**.

User Defined Fields

Exago has optional support for a User Defined Field (UDF) in a report. UDFs allow for storing custom attributes within reports. You can use UDFs in a custom folder management solution in order to associate an ID with an individual report.

Note: UDFs are not required for folder management. They are provided for convenience.

Certain folder management methods are overloaded to support UDFs. These methods are associated with report retrieval. If a UDF is associated with a report, Exago will pass it as an argument to these methods.

If a UDF is set it is stored in an element in the report XML called `<linked_udf>`. Report storage methods do not overload a UDF argument since they include the report XML as an argument.

List of Methods

Note: The methods will use the **parameters** 'companyId', and 'userId' which should be set through the Api as users enter Exago from the host application.

Note: If using a .Net Assembly, the folder management code can use alternative method signatures to be passed Exago's SessionInfo object for additional flexibility. See **Accessing SessionInfo in Folder Management** for more information.

`string` GetReportListXml(`string` companyId, `string` userId)

Description	Returns a string listing folders and report names in xml format (see example).
Remark	For reports set the flag <code><leaf_flag></code> to True. For folders set this flag to False. If an error occurs return null and a generic error will be displayed to the user.
Example	Returns: <pre><entity> <name>Travis' Reports</name> <leaf_flag>>false</leaf_flag> <readonly_flag>>false</readonly_flag></pre>

```

<entity>
  <name>Sales Report</name>
  <leaf_flag>true</leaf_flag>
  <readonly_flag>>false</readonly_flag>
</entity>
<entity>
  <name>Employee Reports</name>
  <leaf_flag>>false</leaf_flag>
  <readonly_flag>>false</readonly_flag>
  <entity>
    <name>Employee Benefits Report</name>
    <leaf_flag>true</leaf_flag>
    <readonly_flag>>false</readonly_flag>
  </entity>
</entity>
</entity>

```

string GetReportXml(string companyId, string userId, string reportName, [string udfData = null])

Description	Returns a string containing the report in xml format.
Remark	If an error occurs return null and a generic error will be displayed to the user.

string SaveReport(string companyId, string userId, string reportName, string reportXml)

Description	Saves a report (reportName is fully qualified)
Remark	Returns an error message if one occurs, else return null. Note: To support multi-language functionality, if the returned string matches the id of any element in the language files then the string of that language element will be displayed to the user instead of the returned value. For more information see Multi-Language Support .

string DuplicateReport(string companyId, string userId, string reportName, string reportXml)

Description	Duplicates a report (reportName is fully qualified). If this method is not provided, SaveReport will be called.
Remark	Returns an error message if one occurs, else return null. Note: To support multi-language functionality, if the returned string matches an id of any element in the language files then the string of that language element will be displayed to the user instead of the returned value. For more information see Multi-Language Support .

string DeleteReport(string companyId, string userId, string reportName, [string udfData = null])

Description	Deletes a report (reportName is fully qualified)
--------------------	--

Remark	<p>Returns an error message if one occurs, else return null.</p> <p>Note: To support multi-language functionality, if the returned string matches the id of any element in the language files then the string of that language element will be displayed to the user instead of the returned value. For more information see Multi-Language Support.</p>
---------------	--

`string` RenameReport(`string` companyId, `string` userId, `string` reportName, `string` newReportName, [`string` udfData = null])

Description	Renames a report (reportName & newReportName are fully qualified). If this method is not provided, DeleteReport and SaveReport will be called.
Remark	<p>Returns an error message if one occurs, else return null.</p> <p>Note: To support multi-language functionality, if the returned string matches the id of any element in the language files then the string of that language element will be displayed to the user instead of the returned value. For more information see Multi-Language Support.</p>

`string` AddFolder(`string` companyId, `string` userId, `string` folderName)

Description	Adds a report folder (folderName is fully qualified). Folder should not be named to one that already exists.
Remark	<p>Returns an error message if one occurs, else return null.</p> <p>Note: To support multi-language functionality, if the returned string matches the id of any element in the language files then the string of that language element will be displayed to the user instead of the returned value. For more information see Multi-Language Support.</p>

`string` DeleteFolder(`string` companyId, `string` userId, `string` folderName)

Description	Deletes a report folder (folderName is fully qualified).
Remark	<p>Exago' default report template management will not allow a folder to be deleted that contains any reports.</p> <p>Returns an error message if one occurs, else return null.</p> <p>Note: To support multi-language functionality, if the returned string matches the id of any element in the language files then the string of that language element will be displayed to the user instead of the returned value. For more information see Multi-Language Support.</p>

`string` RenameFolder(`string` companyId, `string` userId, `string` oldName, `string` newName)

Description	Renames a report folder (folder names are fully qualified). Folder should not be moved to a location where a Folder with a matching name already exists.
--------------------	--

Remark	Returns an error message if one occurs, else return null. Note: To support multi-language functionality, if the returned string matches the id of any element in the language files then the string of that language element will be displayed to the user instead of the returned value. For more information see Multi-Language Support .
---------------	--

`string MoveFolder(string companyId, string userId, string oldName, string newName)`

Description	Moves a report folder (folder names are fully qualified). Folder should not be renamed to one that already exists.
Remark	Returns an error message if one occurs, else return null. Note: To support multi-language functionality, if the returned string matches the id of any element in the language files then the string of that language element will be displayed to the user instead of the returned value. For more information see Multi-Language Support .

`bool ExistFolder(string companyId, string userId, string folderName)`

Description	Determines if a folder exists.
Remark	Returns true or false.

`List<string> GetTemplateList(string companyId, string userId)`

Description	Returns a list of strings or a string array containing the available templates.
Remark	If an error occurs return null and a generic error will be displayed to the user.

`byte[] GetTemplate(string companyId, string userId, string templateName)`

Description	Returns a byte array containing the template.
Remark	If an error occurs return null and a generic error will be displayed to the user.

`string SaveTemplate(string companyId, string userId, string templateName, byte[] templateData)`

Description	Saves a template
Remark	Returns an error message if one occurs, else return null.

	<p>Note: To support multi-language functionality, if the returned string matches the id of any element in the language files then the string of that language element will be displayed to the user instead of the returned value. For more information see Multi-Language Support.</p>
--	---

List<string> GetThemeList(string companyId, string userId, string themeType)

Description	Returns a list of strings or a string array containing the available themes.
Remark	<p>Valid values of themeType: "CrossTab", "Express".</p> <p>If an error occurs return null and a generic error will be displayed to the user.</p>

bool ExistTheme(string companyId, string userId, string themeType, string themeName)

Description	Determines if a theme exists.
Remark	<p>Valid values of themeType: "CrossTab", "Express".</p> <p>Returns true or false.</p>

string GetThemeXml(string companyId, string userId, string themeType, string themeName)

Description	Returns a string representing the theme in xml format.
Remark	<p>Valid values of themeType: "CrossTab", "Express".</p> <p>If an error occurs return null and a generic error will be displayed to the user.</p>

string SaveTheme(string companyId, string userId, string themeType, string themeName, string themeXml)

Description	Saves a theme.
Remark	<p>Valid values of themeType: "CrossTab", "Express".</p> <p>Returns an error message if one occurs, else return null.</p> <p>Note: To support multi-language functionality, if the returned string matches the id of any element in the language files then the string of that language element will be displayed to the user instead of the returned value. For more information see Multi-Language Support.</p>

Accessing SessionInfo in Folder Management

This section only applies to Folder Management using a .Net Assembly.

After adding a reference to WebReportsApi.dll you may gain additional flexibility by replacing companyId and userId with Exago's sessionInfo object in the methods listed above. The sessionInfo object grants access to all of the parameters, configuration, and report information of the Exago session. See **Exago SessionInfo** for more information.

To utilize the sessionInfoObject **replace** the companyId and userId parameters in the method signatures above with **sessionInfo** sessionInfo (see example below).

Note: To use the sessionInfoObject all the methods must be static.

Utilizing the sessionInfo object will allow the folder management code to access much more information about the user and/or Exago. For example the capability to access the sessionInfo object could be used to pass additional parameters to your folder management code such as the preferred language of the user or from which part of the host application they entered Exago.

Note: Passing the sessionInfo object will lock the folder management Assembly. In order to unlock the assembly, either IIS will need to be restarted, or the application pool running Exago will need to be recycled.

The following is an example of the method signature for GetReportXml to utilize the sessionInfo object.

string GetReportXml(**SessionInfo** sessionInfo, **string** reportName)

Description	Returns a string containing the report in xml format.
Remark	<p>If an error occurs return null and a generic error will be displayed to the user.</p> <p>companyId and userId can still be retrieved using the calls sessionInfo.SetupData.Parameters.GetValue('companyId') and sessionInfo.SetupData.Parameters.GetValue('userId') respectively.</p> <p>The sessionInfo object must be the first parameter in the method.</p>