RTI_BRW_GenerateReport

Description

RTI_BRW_GenerateReport is a routine that allows banded reports to be generated with a variety of options.

Syntax

RTI_BRW_GenerateReport(rptFile, rptName, outputName, rptType, overrideListID, rptDetails, bUseGUI, overrideCfg)

Parameters

The function has the following parameters:

Parameter	Description
rptFile	The name of an existing report group <u>or</u> a dynamically generated XML report definition string. ***Note: If the XML report definition string was derived using the READ action of the* RTI_BRWSupport *function, a* Swap \0D0A\ with @FM *command needs to be ran against the XML string. This is because RTI_BRW_GenerateReport looks for the presence of @FM delimiters in* rptFile *to determine if this is an XML report definition string.*** ***If the XML report definition string was derived differently, such as a direct read from the SYSREPOSREPORTS record, then a different step might be required. Newer versions of the BRW embed the* version *into attribute 1 and the* dynamic dictionary data *(or an empty string if there are none) into attribute 2 before the report definition string. These need to be removed from the XML string or the following error will result:*** `Data at the root level is invalid. Line 1, position 1`
rptName	If *rptFile* contains the name of a report group, then *rptName* contains the name of a specific report to run (or "*" to run all the reports in the group). You may also specify a semicolon-delimited list of report names if desired. ***Note: The "*", or wildcard, feature and the semicolon delimited list only work with specific CFG_OIBRW configuration settings. In short, only those settings that create new engines and then close them after each report will work properly. See the* CFG_OI BRW Configuration *article for more information.***
outputName	The path and name of the file to save the output to (if producing a PDF, HTM, etc. document). Leave this blank to generate printed output.
rptType	The type of output to generate, and can be PDF, TIFF, HTML, TEXT, XLS, or XLSX (or PRINT to generate printed output, but that isn't really needed if you leave *outputName* blank).
overrideListID	The ID of a saved list that contains the record keys you want this report generation to use. If there is no *overrideListID* specified, but there *is* an active select list when RTI_BRW_GenerateReport is called, then the active list will be used instead.

rptDetails	When the *rptType* is PDF, you can specify the access permissions for the PDF, including the password(s) for PDF access. If *rptType* is PRINT (or ""), you can use this parameter to specify the printer name to use and other report options (see table below):

Attribute	*rptType*=PDF	*rptType*=PRINT or "" (*bUseGUI* should be 0 or "")
<1, 1>	Owner password	Printer name (as known by the Windows Print Manager)
<1, 2>	User password	Number of copies
<1, 3>	Access permissions - a string with any of the following characters: • *C*=Copying content allowed • *E*=Editing content allowed • *A*=Annotation editing allowed • *P*=Printing allowed	0 or "" for Portrait, 1 for Landscape
<1, 4>	*Not applicable*	<1, 4, 1> = Left Margin (Hundreds of an inch, i.e., 100=1") <1, 4, 2> = Top Margin (Hundreds of an inch, i.e., 100=1") <1, 4, 3> = Right Margin (Hundreds of an inch, i.e., 100=1") <1, 4, 4> = Bottom Margin (Hundreds of an inch, i.e., 100=1")
<1, 5>	*Not applicable*	Page size code
<1, 6>	*Not applicable*	Print Preview caption. **(Only available in OI v10, otherwise, not applicable.)**

For all report types (i.e., *rptType* can be set to anything), <1, 1, 1> can be set to a literal "UID" and <1, 1, 2> can be set to any custom value. Doing this informs the BRW that this report has a unique identifier, which can be used in the INIT or TERM actions of the designated OI_BRW_FILTER hook.

bUseGUI		

Value	Behavior
0 or ""	Prints the report silently to the default printer. (If configured for outputting to a file, the file is generated without any prompting.)
1	If *rptType* is set to empty or "PRINT", a value of 1 shows a Print Dialog for the report and then prints to the specified printer. If *rptType* is set to a supported file type, a value of 1 shows a Configuration Dialog specific to the file type.
2	If *rptType* is set to empty or "PRINT", a value of 2 displays the report in the preview window. If *rptType* is set to a supported file type, a value of 2, it works the same as when *bUseGUI* is set to 1.

overrideCfg	**OECGI Invocation Only** - Override configuration settings to be used for this instance of the BRW. While the format of this setting uses the same attributes as the CFG_OIBRW configuration record, no defaults will be applied. Therefore, each attribute will have to be explicitly populated (i.e., *fully resolved*) with the correct value. See the notes regarding each attribute's default value in the the CFG_OIB RW Configuration article to understand how to resolve these values. The purpose for an override setting is to ensure that systems like O4W can call the BRW with applicable configuration settings. For instance, the default configuration might be to share the engine, but O4W systems require a new connection. ***Note: For OI 9.4 and earlier, even if this is not an OECGI invocation, pass in an empty string to avoid a VNAV error. OI 10 will have a fix for this issue.***

Returns		

Value	Meaning
0	Success
-1	Either parameter "reportFile" or "reportName" is null ("")
1	Report definition is null - specified report is not on file
2	Unable to open SYSLISTS table
<error text>	Error returned by call to the BRW component
<ole status>	Error returned by call to the BRW component

Remarks

Multiple reports, even from different report groups, can be ran from a single **RTI_BRW_GenerateReport** call through @RM delimited values for each parameter. This is similar to the way Set_Property supports @RM delimited parameters. While multiple reports can be specified to export to a file, this cannot be used to create a single file that contains all reports. Each report must have its own file name or older reports will be overwritten.

RTI_BRW_GenerateReport is the only way to create PDF reports without the use of the Presentation Server. Using RTI_BRWSupport or the LOADREPORT **Set_Printer** message both work through OIPI.Net, which requires the Presentation Server.

Dynamically created XML report definitions do not support *Dynamic Dictionaries*. These will come through as plain text.

See the notes for the *rptFile* parameter for important information when passing in dynamically generated XML report definition strings.

See Also

RTI_BRWSupport