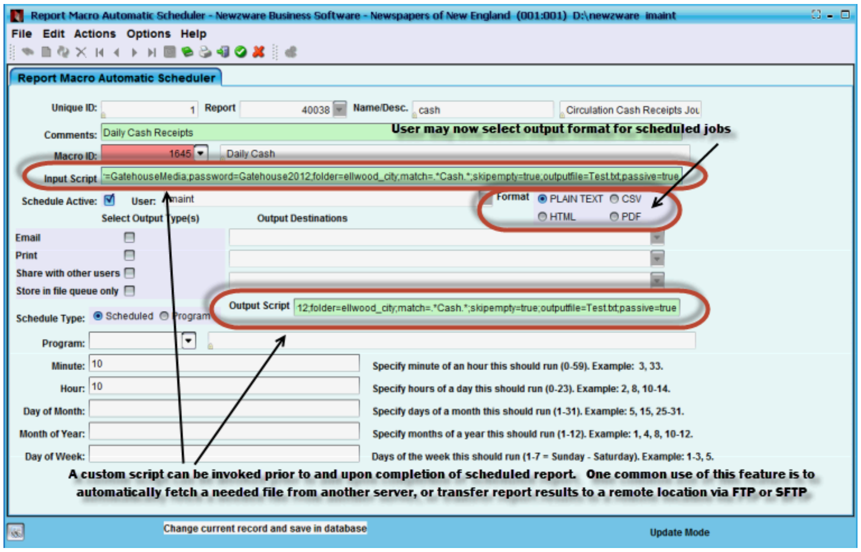
Back to Build Notes

The report scheduler now offers a selection of output format options, as well as the ability to invoke a script prior to and upon report completion.

The output format options are self-explanatory - simply select the desired output format for the scheduled job, as shown below.


Report Macro Automatic Scheduler - Newzware Business Software - Newspapers of New England (001:001) D:\newzware imaint
File
Edit
Actions
Options
Help
Report Macro Automatic Scheduler

Unique ID: 1 Report 40038 Name/Desc. cash Circulation Cash Receipts Jou

Comments: Daily Cash Receipts

Macro ID: 1645 Daily Cash

Input Script =GatehouseMedia;password=Gatehouse2012;folder=ellwood_city;match=.*Cash.*;skipempty=true;outputfile=Test.txt;passive=true

Schedule Active: ☑ User: imaint

Format ● PLAIN TEXT ○ CSV ○ HTML ○ PDF

Select Output Type(s) Output Destinations

Email □

Print □

Share with other users □

Store in file queue only □

Output Script 12;folder=ellwood_city;match=.*Cash.*;skipempty=true;outputfile=Test.txt;passive=true

Schedule Type: ● Scheduled ○ Program

Program:

Minute: 10 Specify minute of an hour this should run (0-59). Example: 3, 33.

Hour: 10 Specify hours of a day this should run (0-23). Example: 2, 8, 10-14.

Day of Month: Specify days of a month this should run (1-31). Example: 5, 15, 25-31.

Month of Year: Specify months of a year this should run (1-12). Example: 1, 4, 8, 10-12.

Day of Week: Days of the week this should run (1-7 = Sunday - Saturday). Example: 1-3, 5.

User may now select output format for scheduled jobs

A custom script can be invoked prior to and upon completion of scheduled report. One common use of this feature is to automatically fetch a needed file from another server, or transfer report results to a remote location via FTP or SFTP

Change current record and save in database

Update Mode

As noted in the snapshot above, a script can be invoked prior to and upon completion of the scheduled report. This can be used for virtually anything, but is expected to primarily be used to fetch needed files or transfer the report results to some remote destination via FTP or SFTP.

The structure of the script field is a script name and then a series of arguments (parameters) to that script; the script name and parameters must be separated by a comma. The parameters are a series of name/value pairs separated by semi-colons. The default script name is **com.icanon.report.DefaultSchedulerScript**. This default script will handle FTP or SFTP transfers automatically. Other capabilities may be added in the future, and other script types may be added as well.

A couple of examples of scripts and their configuration are shown below. This script would send the report result file via FTP to the server at IP address 192.168.1.100, using the user ID **someuser** and the password **mypass**. It would place the report result file in the folder **asubfolder** on the target server.

com.icanon.report.DefaultSchedulerScript,protocol=ftp;server=192.168.1.100;user=someuser;password=mypass;folder=/asubfolder;

This script would send the report result file via FTP to the server at host name: ftp.ftpXYZ.com, using the user ID **someuser** and the password **mypass**. It would place the report result file in the folder **/newzware/PCF** on the target server, in the outputfile: **TEST_ROUTE_LIST**, and only the files that match the pattern: **.*ROUTE_LIST.*** (note the .* (dot star)is a regular expression that matches any text.)

com.icanon.report.DefaultSchedulerScript,protocol=ftp;server=ftp.ftpXYZ.com;user=someuser;password=mypass;folder=/newzware/PCF;outputfile=TEST_ROUTE_LIST.TXT;match=.*ROUTE_LIST.*

Shown below is a table of the available parameters to the DefaultSchedulerScript

Parameter Name	Values	Description	Required?
protocol	**ftp,sftp**	What protocol to use to execute the script	No
server	server name/IP Address	Domain Name or IP address of target server	Yes
port	integer	TCP/IP port to use to make connection	No
user	actual user ID	User ID on target server	Yes
password	actual password	Password for user ID on target server	Yes
folder	folder/directory name	Folder on server in which to deposit files	No
ftpclient	class to use for FTP	Names a java class to use for FTP transport, eg.**ftpclient=com.icanon.util.jftp.GanymedeFTPClient**	No
privatekey	option se for SFTP	Names a full path to a file containing a prive key, eg.**privatekey=E/cygwin64/home/icanon.support/.ssh/id_rsa_custom**	No
outputfile	any file name	Name to store output file as on target server	No
match	any regular expression pattern	matches a pattern in the file name to determine what files to process	No
skipname	any regular expression pattern	matches a pattern in the file name to determine what files to skip	No
typesort	force delivery order by file type	Ensure all files of a certain type are delivered before other file types. List delimited by !. For example, **typesort=pdf!jpg!xml;** will force all PDF fil es to be sent first, then all JPGs, then XML files. Any other file types will be skipped	No
typefolder	folder/directory name, varying by file type	Target folder on target server can vary by file type. List delimited by !, values separated by colons. For example, **typefolder=xml:/xmlfiles/new!txt:/txtfil es/new;**	No
skipempty	**true,false**	True if an empty (zero-byte) file should be skipped (not transferred)	No
copyfolder	folder/directory name	Local folder on Newzware server to which the report result will also be copied	No
emailon	**true,false**	Default setting for this option is **true**. Email settings are controlled by existence of (and values within) jftpemail.properties file. By default email is on and looks for this file. If it does not exist, no email will be sent. If the jftpemail.properties file does exist, the email can be disabled on certain reports by setting this option to false.	No
propertiesfile	Full path and name of properties file	All of the settings above (and any additional needed) can be set in a properties file. The file must be structured as a series of name/value pairs, one property per line, each line of the form propertyname=propertyvalue.	No
gzip	**true,false**	Files will be compressed to the GZIP compression standard and the file will be given a .gz suffix.	No
zip	**true,false**	Files will be compressed to the ZIP compression standard and the file will be given a .zip suffix.	No
delay	**integer**	Number of milliseconds (20000 is the maximum)	No
action	**put,get**	Controls whether the action is to **get** or **put** (default). The **get** action is useful to fetch a file before launching the report in the now supported Input Script, report schedule section	No
put		The default action - implied	No
get		The optional action - must me explicity named. Other related keywords for get: **pre_folder**, **inputfile**, **outputfile**.	No
inputfile	***filename***	The optional file name where the **get action** will fetch from the remote server.	No
pre_folder	***folder_name***	The optional folder name on the remote server where the **get action** looks for the **inputfile**.	No
fetchfolder	***folder_name***	The optional folder name on the remote server where the **get action** looks for multiple files to retrieve. Anything that matches the **match**.	No
rtf	***folder_name***	Stands for RenameToFolder. The optional folder name on the remote server where the **get action** will move/rename files it retrieves into.	No
passive	**true,false**	If true, the ftp session will enter local passive mode, which can help with certain firewalls restricting anonymous data connections	No

A sample, functional get command follows:

com.icanon.report.DefaultSchedulerScript,protocol=sftp;server=192.168.1.1;user=obits;password=****;pre_folder=home;rtf=/home/nw-processed;
action=get;match=*.*;folder=E:/newzware/ad/data/obitimport/;ftpclient=com.icanon.util.jftp.GanymedeFTPClient;**

What the above command would do is log into the server as user obits; look in the **home** folder designated by the pre-folder tag; **get** everything in that folder that **match**es the *.* tag; place it on the local server in the folder designated by the **folder** tag;and on the FTP server, rename the retrieved file into the folder designated by the **rtf** tag.

The keyword **<datestamp>** is now available to embed the current date in the format: **YYYYMMDD**. This is similar to the keyword **<timestamp>** that includes both the date and the time in the format: **YYYYMMDDhhmmss**.
In addition there are: **<datestamp!>** keyword prefixes that introduce a customizable datestamp using the Simple Date Formatting characters in the Java API: Simple Date Format API where you can customize how a date is displayed. But that's not all. You can add or subtract days, weeks, months or years from the current date using a formula in paranthesis such as: **(-4d)** for 4 days in the past, or **(+1d)** for 1 day in the future. If the current day of the week is Monday, and I want the job to name a file with the text **"Wed"** in it, I would use: **<datestamp!EEE(+2)!>**.

Use of these keywords can be used with the **outputfile** directive that may include both literal text and the replacement keywords. If sending multiple files, the output directive may be limited to a particular file type by using of the **filetype** option to the **outputfile** directive. For example, if sending an XML file and some associated JPGs, but only the XML file is to be renamed, adding **<filetype!xml!!>** to the output file name will cause only the XML file to be renamed (the file type directive itself will not be included in the resulting file name).
A sample usage of this directive follows:

com.icanon.report.DefaultSchedulerScript,protocol=ftp;server=192.168.1.1;user=myuser;password=*******;outputfile=thetargetfilename<datestamp><filetype!xml!!>.xml;skipempty=true**

Back to Build Notes