

Creating Reports using the UC4.DB Reporting Tool Utility

Reporting provides important information on any business processes. UC4 provides a Utility that allows you to easily create and execute reports on information in the database.

The DB.Reporting Utility can be used to create the following types of reports:

- Details of Object Contents
- Forecasting Future Executions
- Statistical Details on Past Executions

Some examples of requests:

- What jobs ran on a specific day
- How many jobs failed with an ENDED_NOT_OK status
- How many JOBS objects are in a specific Client
- What Objects have Agent x defined
- What Jobs are expected to run x days in the future

Steps to creating a Report:

1. Create a .xml Query file using the Reporting GUI
2. Run the DB.Reporting Tool Utility (in Batch mode using the .xml query file)

The formats CSV and HTML are available. A style sheet of your choice can be specified for HTML, which is then integrated in the report file.

To automate the creation of reports, a UC4 job can be created to run the Reporting Utility commands, passing in various values through the use of PromptSets.

More information can be found:

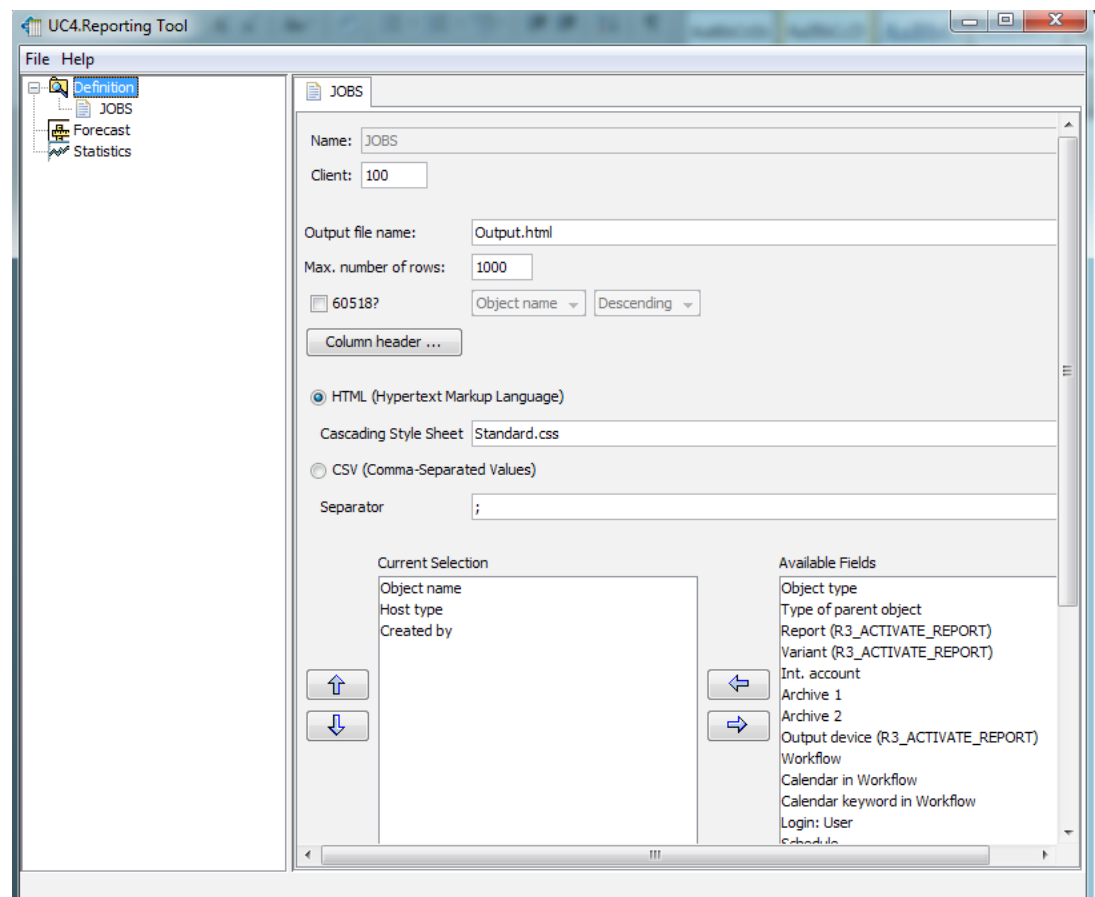
Administration Guide > Utilities > UC4.DB Reporting Tool > Task Evaluations

Step 1 – Creating the .xml Query File

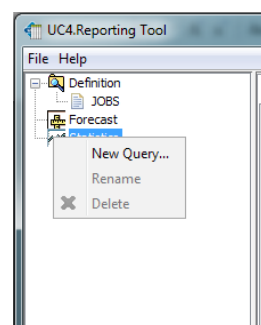
This utility's graphical interface is available for UNIX and Windows. The way it is called depends on the OS you use:

- UNIX: Start the Reporting Tool's interface via the Java Application Launcher using a console command. The required file ucyrepg.jar is available in the utility/bin folder.
- Windows: Under Windows, you can start the interface by using the file "ucyrepg.exe" or "ucyrepg.jar". They are both stored in the utility/bin folder.

Note for Windows: A 32-bit Java version is required if you start the Reporting Tool's interface via the file "ucyrepg.exe". This limitation does not apply if you use the file "ucyrepg.jar".



Right-Click on the type of query you would like to create and select 'New Query'



Step through the Query creation Wizard.


Example:

The following example shows a query that returns a list of all a client's Jobs that use a particular host (WIN01) in the evaluation report.

1. Create a new query, assign a suitable name and select a client.
2. Select the following fields that should be displayed in the output file: object name, object type, host and host type.
3. Selection criteria: After selecting "Host" as a field in the evaluation report, the filter "Object type corresponds to JOBS" is automatically inserted. You can select a particular host by adding the criterion "Host corresponds to WIN01".
4. Determine a name for the output file and the maximum number of lines.

The utility stores your queries in the subfolder "queries" in the form of XML files. The file endings depend on the area that has been queried:

- *.d.xml - definition
- *.f.xml - forecast
- *.s.xml - statistics

| Name | Date modified | Type | Size |
|--|------------------|--------------|------|
|  JOBS.d.xml | 4/7/2013 3:03 PM | XML Document | 1 KB |

More information can be found:

Administration Guide > Utilities > UC4.DB Reporting Tool > Graphical Interface of the Reporting Tool

The Reporting GUI is not dependent on any other component. Its files can be packaged in a .zip file and give to requestor so they can build their own Queries.

The .xml file created can then be moved to where the UC4.DB Reporting Tool Utility will be executed.

The files required would be:

- ucyrepg.exe
- ucyrepg.jar
- ucyrepgp.zip

The first time running the GUI, the ucyrepgp.zip is unpacked and a 'queries' folder is created.

Step 2 – Run the UC4.DB Reporting Tool in Batch mode

Syntax for running Reporting Tool:

UCYBDBRT - Xquery file [-Cclient -IINI file-Llanguage -Rreference date -S-Output file-Tfile type]

Example:

UCYBDBRT -C0100 -XC:\UC4\Utilities\bin\queries\JOBS.d.xml -OC:\UC4\Utilities\reports\JOBS.csv

The database which the Query should be against is defined in the INI.

If no Output directory is defined as a parameter, the report file is created in the Utility/bin directory under the named defined in the Query file.

If reports are viewed on a website like SharePoint, a shared path could also be defined so a report can be posted directly to the website.

Example of a CSV report:

| | A | B | C | D |
|----|--|-------------|----------------------------|----------|
| 1 | Object name | Object type | Status | Runtime |
| 2 | MM.GET.FILES | SCRI | ENDED_OK - ended normally. | 00:00:54 |
| 3 | MM.GET.FILES | SCRI | ENDED_OK - ended normally. | 00:00:45 |
| 4 | MM.GET.FILES | SCRI | ENDED_OK - ended normally. | 00:00:53 |
| 5 | DB.ADMIN | JOBS | ENDED_OK - ended normally. | 00:15:05 |
| 6 | DB.USE | JOBS | ENDED_OK - ended normally. | 00:08:22 |
| 7 | MAWI.GROUP | JOBG | ENDED_OK - ended normally. | 00:00:01 |
| 8 | MAWI.CALL.DAY | CALL | ENDED_OK - ended normally. | 00:20:30 |
| 9 | MAWI.START.SCHEDULE | JSCH | ENDED_OK - ended normally. | 72:01:54 |
| 10 | SMITH/UC4 | USER | ENDED_OK - ended normally. | 00:00:00 |
| 11 | REPORT(C:\UC4\server\TEMP\OAAFKYHR.TXT;0002477063) | REPORT | ENDED_OK - ended normally. | 00:00:10 |
| 12 | REPORT(C:\UC4\server\TEMP\OAAFKYHT.TXT;0002477065) | REPORT | ENDED_OK - ended normally. | 00:00:15 |