**RunQualiJob**

Wednesday, December 24, 2014

10:01 AM

**Supported version**: CloudShell 6.0 or higher

The RunQualiJob EXE enable queue custom CloudShell Job using Command Line Interface.

**EXE inputs**

The RunQualiJob EXE has 1 input:

1. Path to the job JSON file - please see below\*

For example possible way to run the EXE is - from CMD type:

RunQualiJob.exe "C:\Temp\JobJSON.txt"

**RunQualiJob EXE configuration file**

Locate the ‘RunQualiJob.exe.config’ XML file – this file located next to the EXE - under the same directory.

Open the file and configure the following:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Key** | **Value (example)** |
| TestShell server address | TestShellServerAddress | 192.186.80.80 |
| TestShell server port | TestShellServerPort | 8080 |
| The directory of the EXE log | LogFileDirectory | C:\RunQualiJobDirectory |
| TestShell user | TestShellUser | admin |
| TestShell password | TestShellPassword | admin |
| TestShell domain | TestShellDomain | Global |

**Triggered by outlook mail**

As any EXE RunQualiJob could be launched manually\batch file.

Another option to launch RunQualiJob is via outlook mail rule- each time inbox mail (which match specific condition) - the outlook rule would launch and commit the following steps:

1. Save the outlook message to specific directory as \*.msg file.
2. Run bat file – the bat file would execute the RunQualiJob, as part of the EXE execution it would load the \*.msg file. The EXE can also use the mail parameters for the Job execution parameters - for example, define the name of the Job – the RunQualiJob would use the email subject and define the name of the Job accordingly.

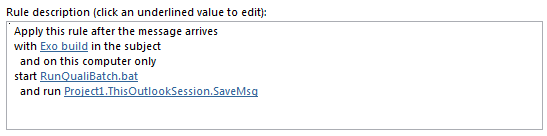
In order to launch this option open the ‘RunQualiJob.exe.config’ XML file and set the following:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Key** | **Value** |
|  | TriggerByMail | True |
| \*.msg emails directory | EmailDirectory |  |

Incase TriggerByMail = True 🡪 an outlook rule must be applied.

The rule must save the mail first as \*.msg file and run the bat file which launch the RunQualijob EXE.

* Save income mail - <http://www.slipstick.com/outlook/archive-outlook/save-incoming-messages-hard-drive/>
* Run bat file – in the rule definition select ‘Start Application’ and point to the bat file. The bat file should contain the RunQualiJob input parameter.



**Wait for job ends**

Once the EXE queue the Job - the EXE can finish its execution – this is the default behavior.

In case the user wants the EXE to wait until all the Job tests finish them execution - he can configure it in the ‘RunQualiJob.exe.config’ file.

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Key** | **Value** |
| Wait For Job Ends | WaitForJobEnds | True |

Incase WaitForJobEnds = True 🡪 the Job tests results would be written in the log file as well as in the EXE application console.

**Path to the JSON file\*:**

CloudShell QualiAPI enable queueing custom job via REST API – RunQualijob EXE use QualiAPI in order to queue the jobs.

The REST API uses JSON object in order to transfer the Job information - for additional information about the JSON structure - open the attached document - 'Quali REST API.html ' - search for the 'Enqueue Custom Job section.

2 Examples of JSON Job representation - the first send tests + test parameters, as the seconded present sending topology information:

{

"Name":"SampleJob",

"Description":null,

"ExecutionServers":null,

"LoggingProfile":"MeasurementsAndResults",

"EstimatedDuration":4.0,

"StopOnFail":false,

"StopOnError":false,

"Tests":[

{

"TestPath":"TestShell\\Tests\\Shared\\SnQ\\Pass",

,"EstimatedDuration":2.0,

"Parameters":[

{

"ParameterName":"PassVar",

"ParameterValue":"Value1",

},

{"ParameterName":"NumericInput",

"ParameterValue":"4"}

],

},

{

"TestPath":"TestShell\\Tests\\Shared\\SnQ\\Fail",

,"EstimatedDuration":2.0,

"Parameters":[

{

"ParameterName":"FailVar",

"ParameterValue":"ddd"

}

],

}

]

,"Topology":null,"DurationTimeBuffer":2.0,"EmailNotifications":"SuiteAndErrors","Type":"TestShell"};

------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

{

"name": "job1",

"description": null,

"executionServers": [],

"loggingProfile": "Results",

"estimatedDuration": 123,

"stopOnFail": false,

"stopOnError": false,

"tests": [

{

"TestPath": "TestShell\\Tests\\Shared\\SnQ\\Pass",

"TestDuration": "1",

"Parameters": []

},

{

"TestPath": "TestShell\\Tests\\Shared\\SnQ\\Fail",

"TestDuration": "1",

"Parameters": []

}

],

"Topology":{"Name":"CITopologyName","GlobalInputs":[],"RequirementsInput":[],"AdditionalInput":[]},

"durationTimeBuffer": 10,

"emailNotifications": "All",

"type" : "TestShell"

};