Ric Generator (Framework) Spec

# Overview

Ric Generator is a window form tool used for grabbing and parsing information from the download data from exchange web site. It’s an execution file. Major features by now:

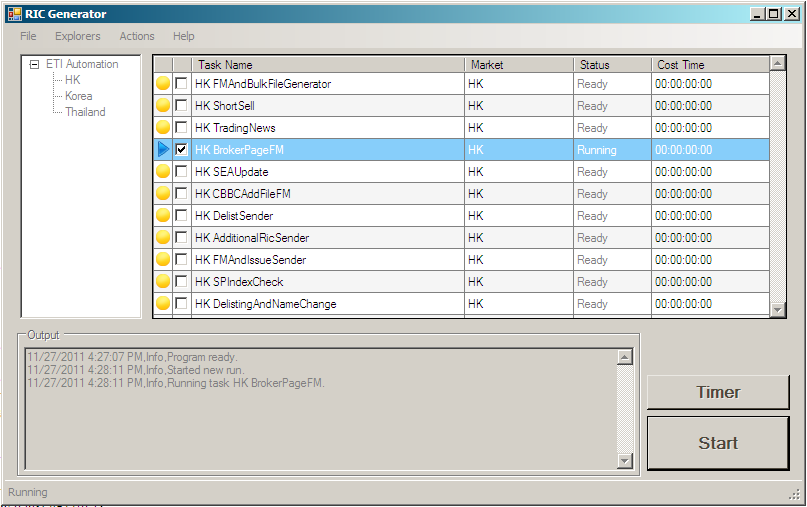
* Tree-model task organization;
* Task scheduling;
* Task configuration;
* Task result viewer;
* Task status tracking;
* Database table viewer

# Feature Details

## Tree-model task organization

Market as the branch. User can select all the tasks in a specified market, that is, if user select a specified market branch, only tasks of this market will be displayed.

For each task, there’re “Task Name”, “Market”, “Status” and “Cost Time”.



*Figure 1: Tree-model task view*

* Task Name: Distinguish each task, it was named according to the task requirements.
* Market: Indicate which market the task belongs to.
* Status: Indicate the status of the task, it has four statuses: Ready, Running, Completed and Failed. Accordingly, the first icon column can also indicate the task status. Status will be updated in real time during the running process.
* Cost Time: Its value specifies how much time has cost for running this task. By default, the value is “00:00:00:00”

## Task scheduling

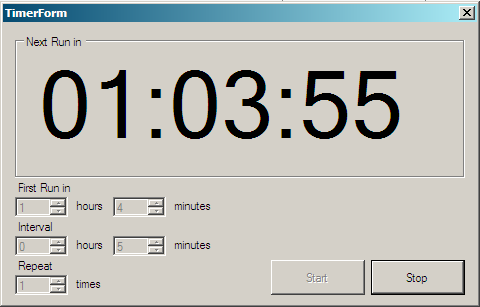
There’re two scheduling mode: Immediate mode and Timer mode

* Immediate mode: User can immediately execute tasks after specifying tasks by clicking “Start” button (Figure1: Tree-model task view).
* Timer mode: User can choose this scheduling mode by clicking “Timer” button (Figure1: Tree-model task view). After clicking “Timer” button, “TimerForm” (Figure 2: Task scheduling) will pop up. User can set three kind parameters: First Run, Interval and Repeat.

First Run in: Tasks will be execution after the time value;

Interval: The value specifies the time interval between each running time;

Repeat: The number of times the tasks will be run.



*Figure 2: Task scheduling*

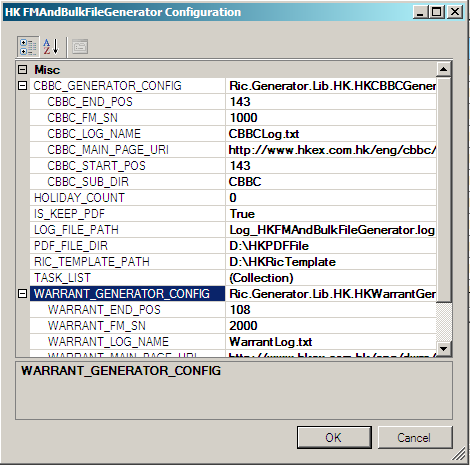
After setting this parameters, start countdown by clicking “Start” button. User can see how much time left for the next run. User can also stop the countdown by clicking “Stop” button.

## Task configuration

Each task has its own configuration editor to update parameters (Figure 3: Configuration editor).

On the task view form (Figure1), after specifying a task, user can double click the task or right click and select “Open config”, configuration form will pop up. Users can update parameters on this panel.

And user can click “OK” to save the modifications, click “Cancel”, the changes will be dropped.



*Figure 3: Configuration editor*

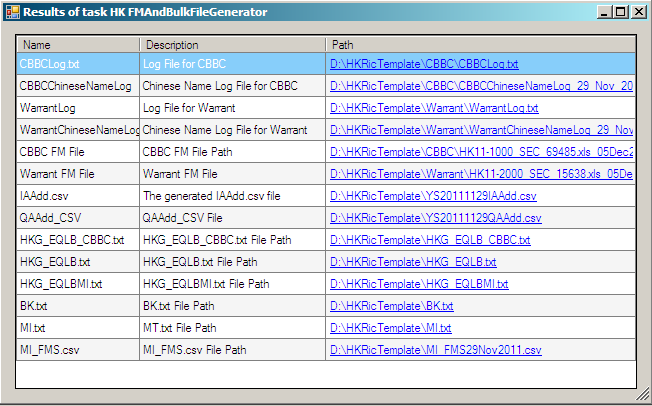
## Task result viewer

After the execution finished, user can view the results on “Result View Form” (Figure 4) by right clicking and selecting “Open Result”.

For each result for a task:

* Name: The result name which can be a file, directory and so on.
* Description: The description for this task which will give detailed description information.
* Path: The path value for the result, usually a file path, directory or Uri.

User can specify a result and click, then the related file, webpage will be opened. User can check and modification.



*Figure4: Result View Form*

## Task status tracking

For each task, there’s “Status” property (column 5, Figure 1). There’re 4 statuses during a task running process: Ready, Running, Completed, Failed. The task status can also be identified on the first icon column (Figure 1)

|  |  |  |
| --- | --- | --- |
| Icon | Status | Description |
|  | Ready | The task is ready to be run |
|  | Running | The task is in the running process |
|  | Completed | The task has finished successfully getting the result successfully |
|  | Failed | The task has finished with some errors |

And after the task execution finished, user can get the cost time from the last column (Figure 1).

In the output text box (Figure 1), real time statistical data will be displayed: the start running time of each task, which task is running, and the error message if a task is failed.

## Database Viewer