MR Tool design doc

目录

Overview

- **What’s MR**

  MR or “Market rehearsal” is the system test between brokers and HKEx for new exchange functions.

- **Why is this tool needed**

  Brokers are required to use the production environment for the test, and the environment need to be resumed after the test. So to avoid the human error in the deployment and environment resume procedures, this tool is developed.

- **What’s this tool for**

  This tool will execute below steps in the expected MR flow:

  1. Team create MR Info file (xml, MUST) and MR package (zip, Optional) through MR Tool, then send them to client
  2. Client/System support open MR tool, import MR Info file (xml) and MR package (zip, if has) into it
  3. MR tool finishes the deployment and MR tasks scheduling
  4. When meeting issues during MR, Client can manually run “restart” for simple fix through MR tool
  5. After MR, MR tool generates settle files and resume the environment

- **Import files for a MR, includes:**
  - MR info file (.XML, MUST): include MR basic info (root folder, name, start time, end time, notice), deploy tasks and MR tasks
  - MR package(.Zip, Optional): include the servers, configuration files etc. for the MR

- **Platform and programming language**

  C#.net

Elements in this tool

- **State**: to identify MR’s state, the state information would show in UI and so as the instructions under specific state. Also, under different states, there are different actions available in UI. States include:
  - Initial state
  - Deployed state
  - Deploy Error state
  - Doing MR state
  - Startup Error state
  - Finished state
  - Finish Error state

- **Task**: represent a program (.exe/.cmd/.bat). There are two kinds of task:
  - Deploy tasks: should be done in the deploy steps
  - MR tasks: execute after deployment to provide MR environment, and resume normal environment
  - How to define a Deploy task or a MR task, please refer to MR info file (.xml) introduction

- **Execute Unit**:
  - Belong to MR tasks, the unit of execution. There are two kinds of execute unit: start unit and shutdown unit.
    - If a MR task is an .exe, then it has two execute units: one for starting, one for shutting down (Of course, if an .exe can shut down by itself, there would be only one start unit).
    - If a MR task is a .bat or a .cmd, then there would be only one start unit
  - Each execute unit has an execute time. For a start unit, it’s the start time of its MR task; and for a shutdown unit, it’s the end time of its MR task.
  - In the deployment, all the execute units would be set a timer for scheduling If its execute time< now, else wouldn’t, means the execute unit is expired for scheduling.
  - There are two lists of execute units:
    1. All execute units list
    2. Expired execute units list, when an execute unit’s execute time is expired, it would be added to this list

- **Actions**: for clients manually take by click buttons in UI, include:
  - Deploy
    - Steps:
      a. Get MR info from MR info file, include:
         1. MR root folder path
         2. MR name

3. MR start time and end time
4. Notice message

b. Create MR folder in root folder, folder name = MR Name + Date
c. Copy MR Info file and MR package to MR folder
d. Unzip MR package to MR folder
e. Get Deploy Tasks from MR info file and start to execute them by sequence
f. Get MR tasks and set schedules for them

- If any steps in deploy a-f steps meets error, all the steps that after it wouldn't execute
- If a-f steps are all successful, DeployedMRFolderPath, DeployedMRConfigPath, DeployedMRPackagePath will write into Config.ini which is under the program folder
- If the MR tool is been shut down after a deployment, then when restarting MR tool, it would load the deployed MR from the information in Config.ini

◆ Restart: execute all the units in Expired execute units list by execute time sequence. Execute means shut first (if shut available and the MR task of the unit has configured doRestart = Y), then start.
◆ Finish: execute the units from all execute units list whose schedule is not executed yet.
◆ Clear: to clear the existing deployment. Include a "Finish" action in the steps

- **How to determine an Execute Unit executes successfully**
  - If its MR task is an .exe, MR tool can't detect. Need to handle by a check task.
  - If its MR task is a .cmd/.bat, by checking the %errorlevel% code returned, if it's 1, means failed, if it's 0, means successfully.
  - Add a check task (script) to check the program runs fine, the check task should differ to different MR tasks. In addition, the check task must return its checking result as %errorlevel% so that MR tool can determine if programs running fine.

MR info file (.xml) introduction

```
<?xml version="1.0" encoding="utf-8"?>
<Info>
    // MR basic information, surrounded with <basicInfo>
  <basicInfo>
    <MRName>MRTest</MRName>
    <RootFolder>C:\oms-bin-rehearsal</RootFolder>
    <Date>2015-06-11</Date>
    <StartTime>19:27:35</StartTime>
    <EndTime>23:27:35</EndTime>
    <Notice>a test</Notice>
  </basicInfo>
    // Deploy tasks, surrounded with <DeployTasks>, and each task surrounded with <task>
  <DeployTasks>
    <task>
      <name>D1</name>
      <IsInPackage>Y</IsInPackage>
      <path>feedExRate.cmd</path>
     //If the execute file in the MR package, Y/N, if choose "N", means didn't provide the execute
     file, then can configure an URL to locate the specific file in client's server, e.g.
     <IsInPackage>N</IsInPackage>
      <path>c:\oms-bin\feedExRate.cmd</path>
      <sequence>1</sequence>   // the execute sequence of this deploy task in all the deploy tasks
      <executeDuration>10</executeDuration> // how many seconds the task need to execute
    </task>
    <task>.....</task>
    <task>.....</task>
  </DeployTasks>

    // MR tasks, surrounded with <MRTasks>, and each task surrounded with <task>
  <MRTasks>
    <task>
      <name>MR2</name>
     //If the execute file in the MR package, Y/N, if choose "N", means didn't provide the execute
     file, then can config an URL to locate the specific file in client's server.
      <IsInPackage>Y</IsInPackage>
      <path>Dds12.exe</path>
     // you can config a port for the execute file (can be null), then when shut the program, would shut
     the program that use the port. If port not configured, then shut the program by execute file name
      <port>9101</port>
      <startTime>15:33:19</startTime>
     // endTime can be null (not set), means this is a .cmd/.bat file, no need to shut down
```

```
    <endTime>21:33:19</endTime>
   // how many seconds does this program take for starting
    <startDuration>5</startDuration>
   // if restart the program if client manually click the restart button
    <doRestart>Y</doRestart>
  </task>
  <task>
    <name>MR3</name>
    <IsInPackage>Y</IsInPackage>
    <path>feedExRate.cmd</path>
    <port></port>
    <startTime>18:34:11</startTime>
    <endTime></endTime>
    <startDuration>10</startDuration>
    <doRestart>Y</doRestart>
  </task>
  </MRTasks>
</Info>
```

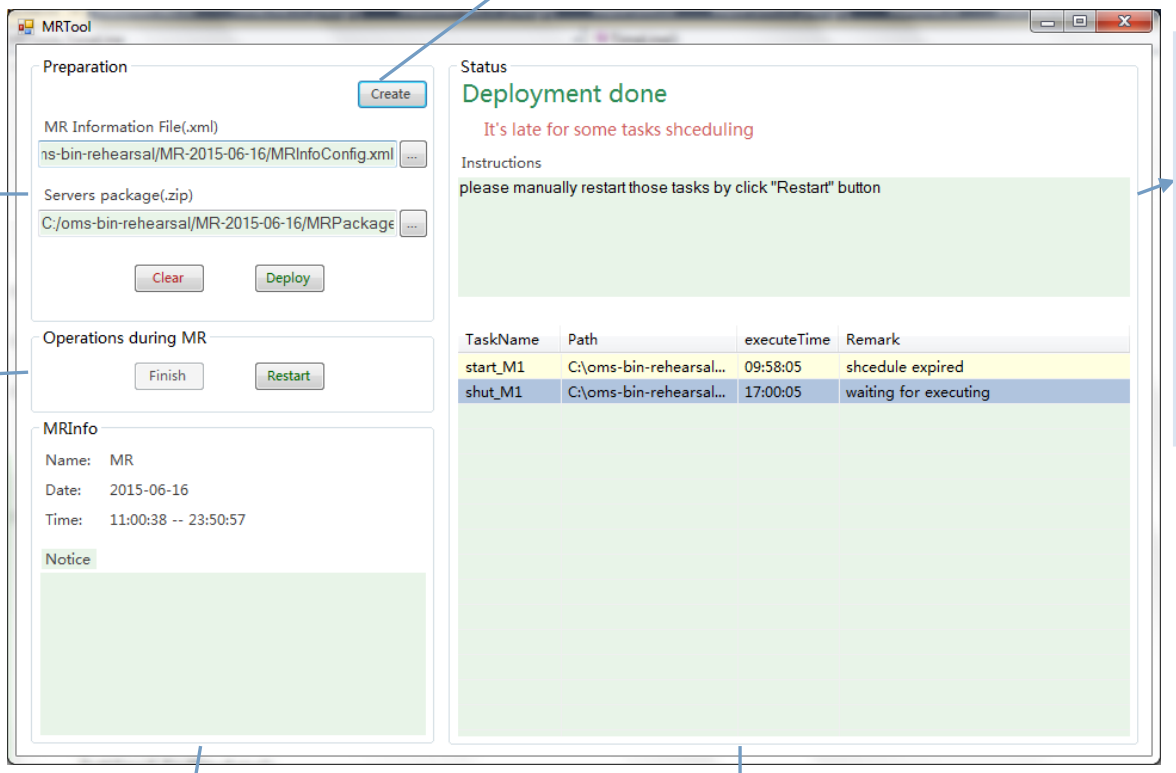
Main Form UI introduction

Import MR Information file and package. By clicking “Clear” to clear, by clicking “Deploy” to deploy

Create MR information file (xml) and package (zip)

Status and instruction zone. To tell client what’s the MR state and under this state what actions they can take

Operation zone that Client can do when during MR. Include: Finish and Restart



MR information, read only

Execute details, and there are there are 4 different status indicated by back color.

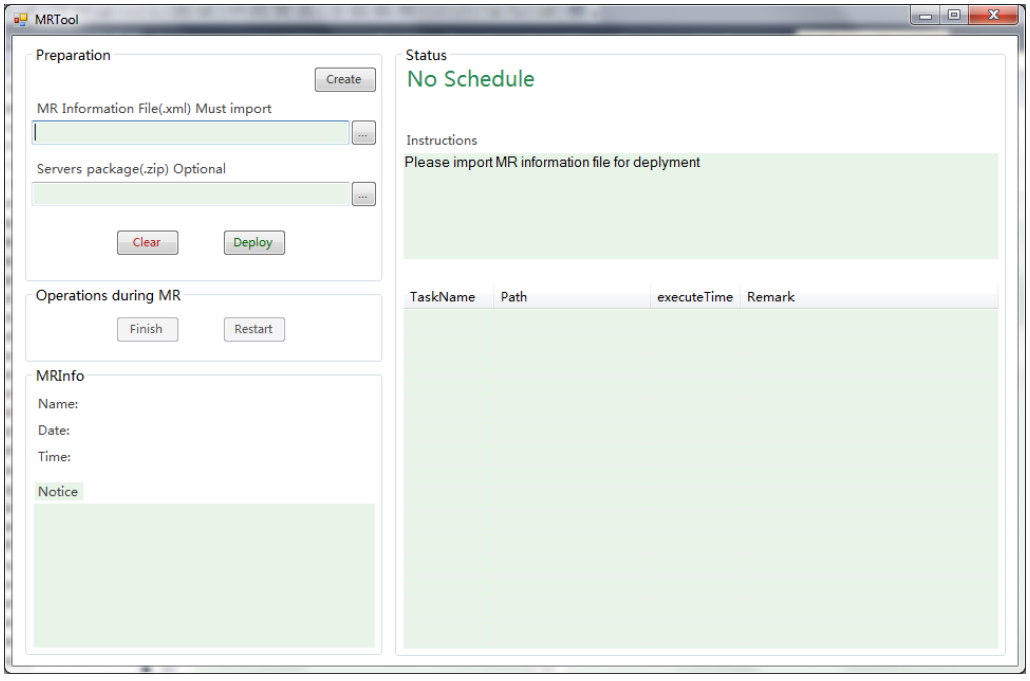
1. Green, means execute successfully
2. Yellow, means the schedule of the unit is expired
3. Blue, means the schedule set successfully and waiting for executing
4. Red, means execute failed

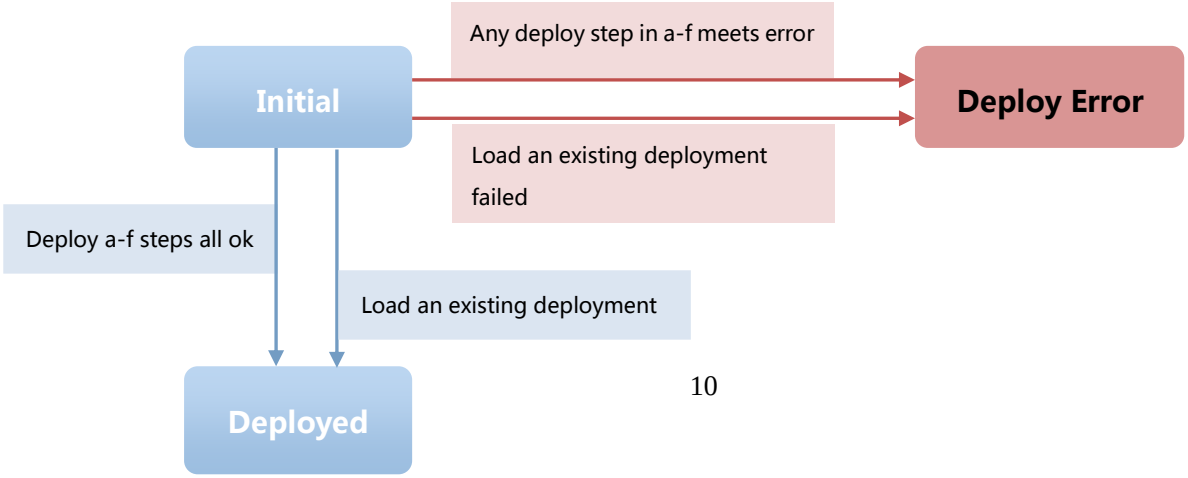
States transfer flow

States introduction

Initial state

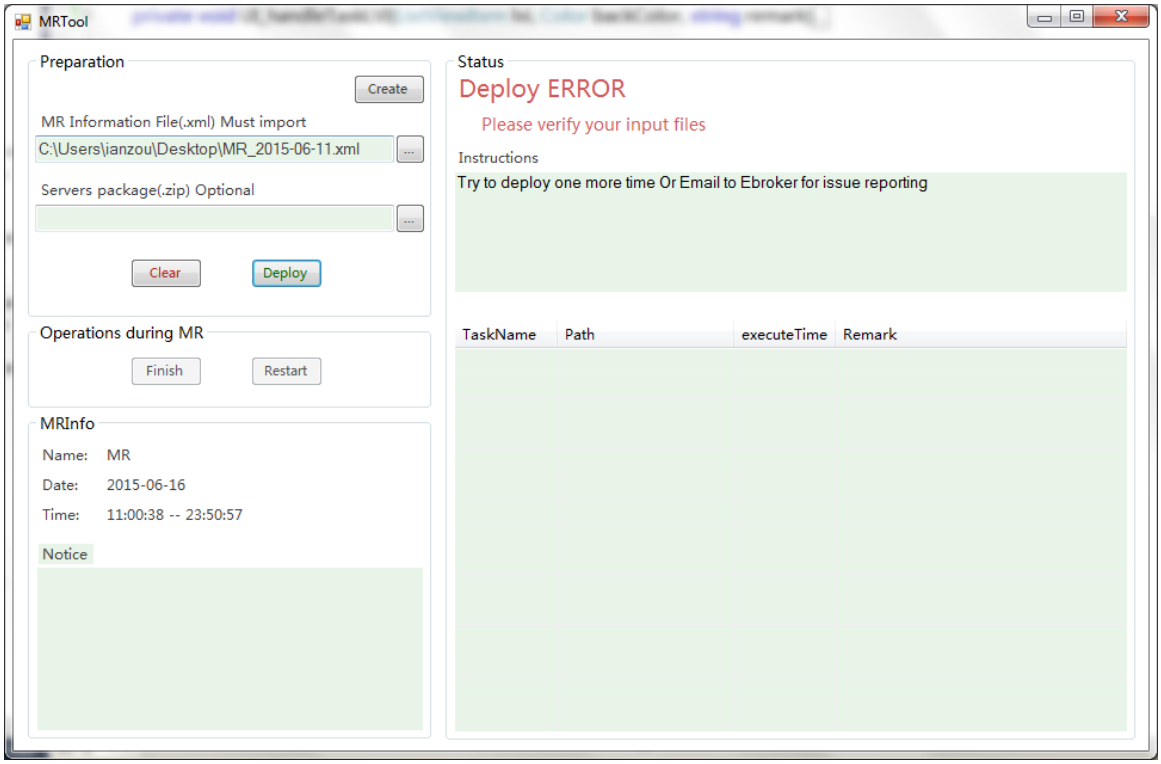
- When open the MRTool program, state initialized to be Initial state
- In this state, two actions can take: Deploy, Clear(Finish and Restart button are not available in UI)
- When deploy, need import MR info XML file(must) and MR package file(optional), if not provide the package, then suppose all the execute file in MR info file can be find in client’ s server
- If there is already a deployment, the deployment info would store in config.ini, when restart program, would load the deployment info.
- If any steps in deploy a-f steps meets error or load the deployment info failed, state would turn into “Deploy Error” state
- If all a-f steps meets successfully or load the deployment info successfully, state turn into “Deployed” state

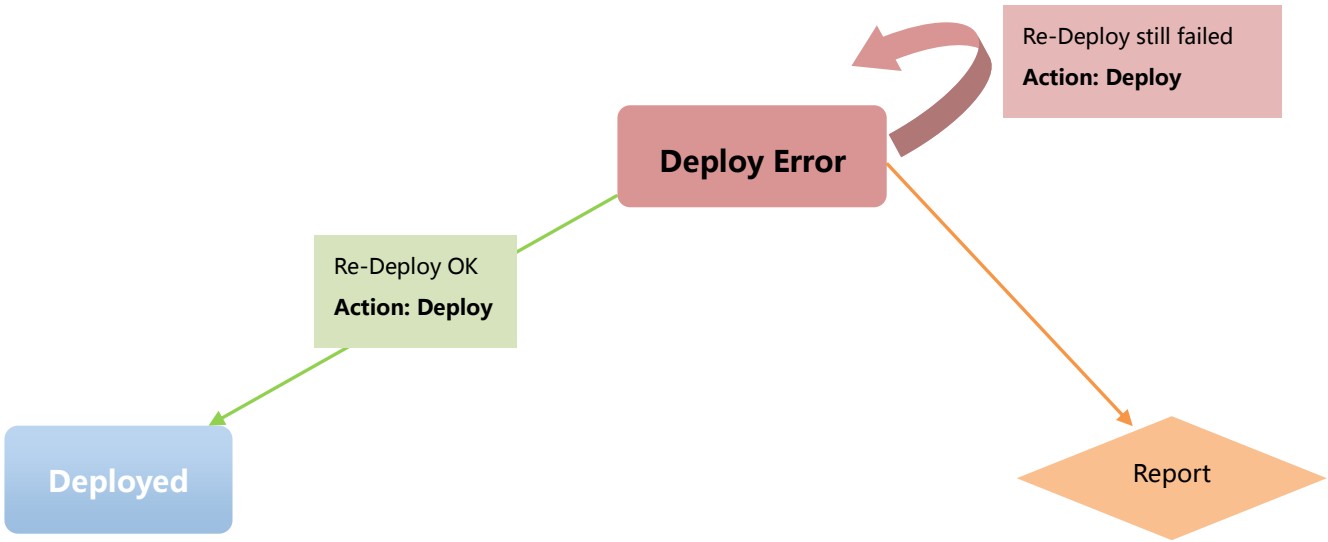




Deploy Error state

In this state, two actions can take: Deploy, Clear (Finish and Restart are not available in UI)

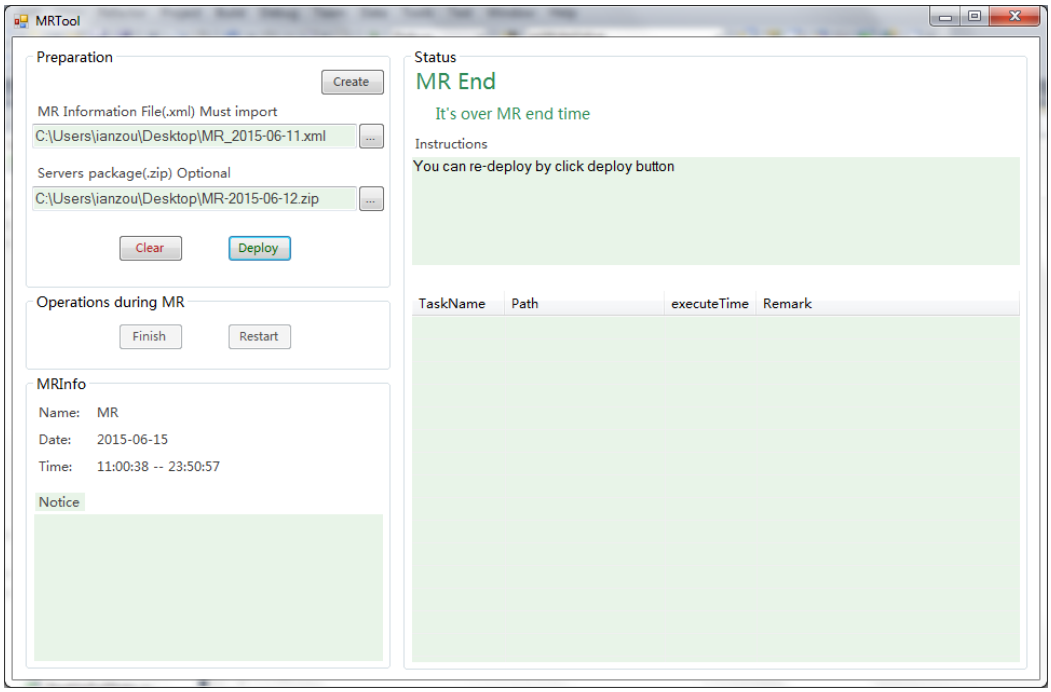




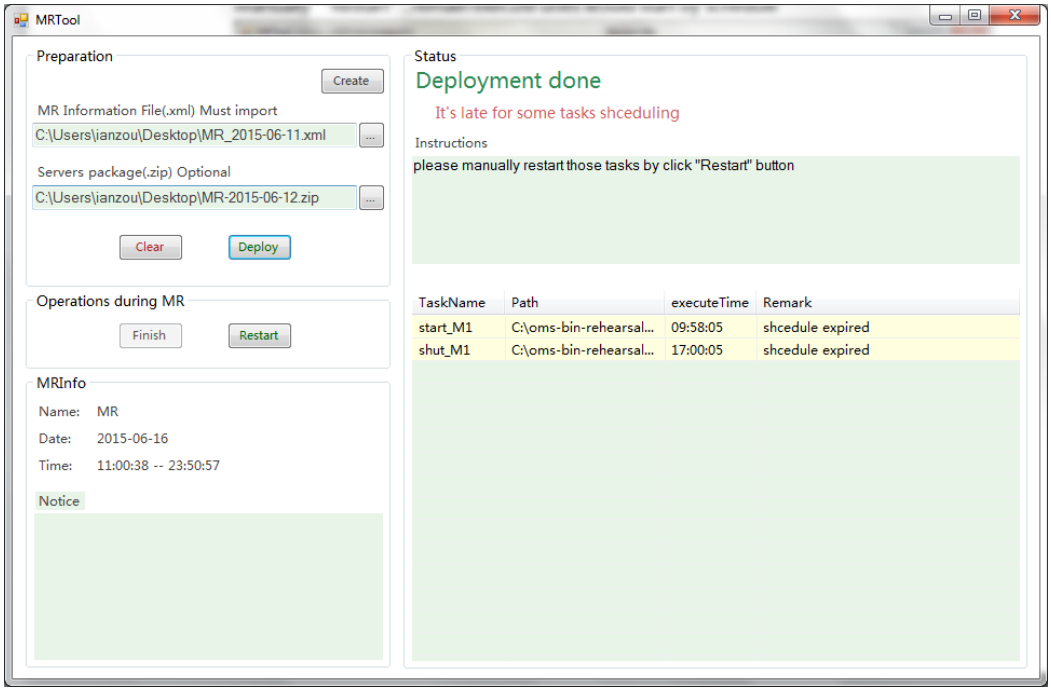
Deployed State

In this state, three actions can take: Deploy, Clear and Restart (if need)

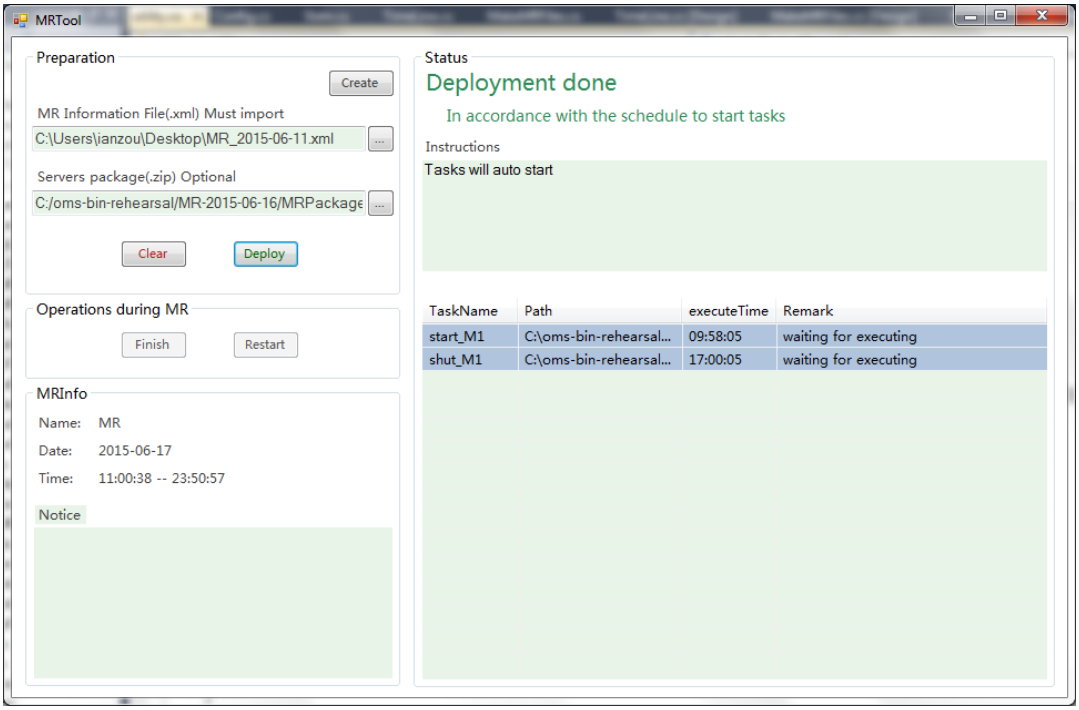
- Client can choose to Re-Deploy by taking action: Deploy**. When Deploy, program will clear the MR info in UI and Finish execute units first.**
- If MR end time<Now, will indicate the MR is over, client can choose to re-deploy.
  For below screen dump, now is 2015-06-16, and MR end time is 2015-06-15 23:50

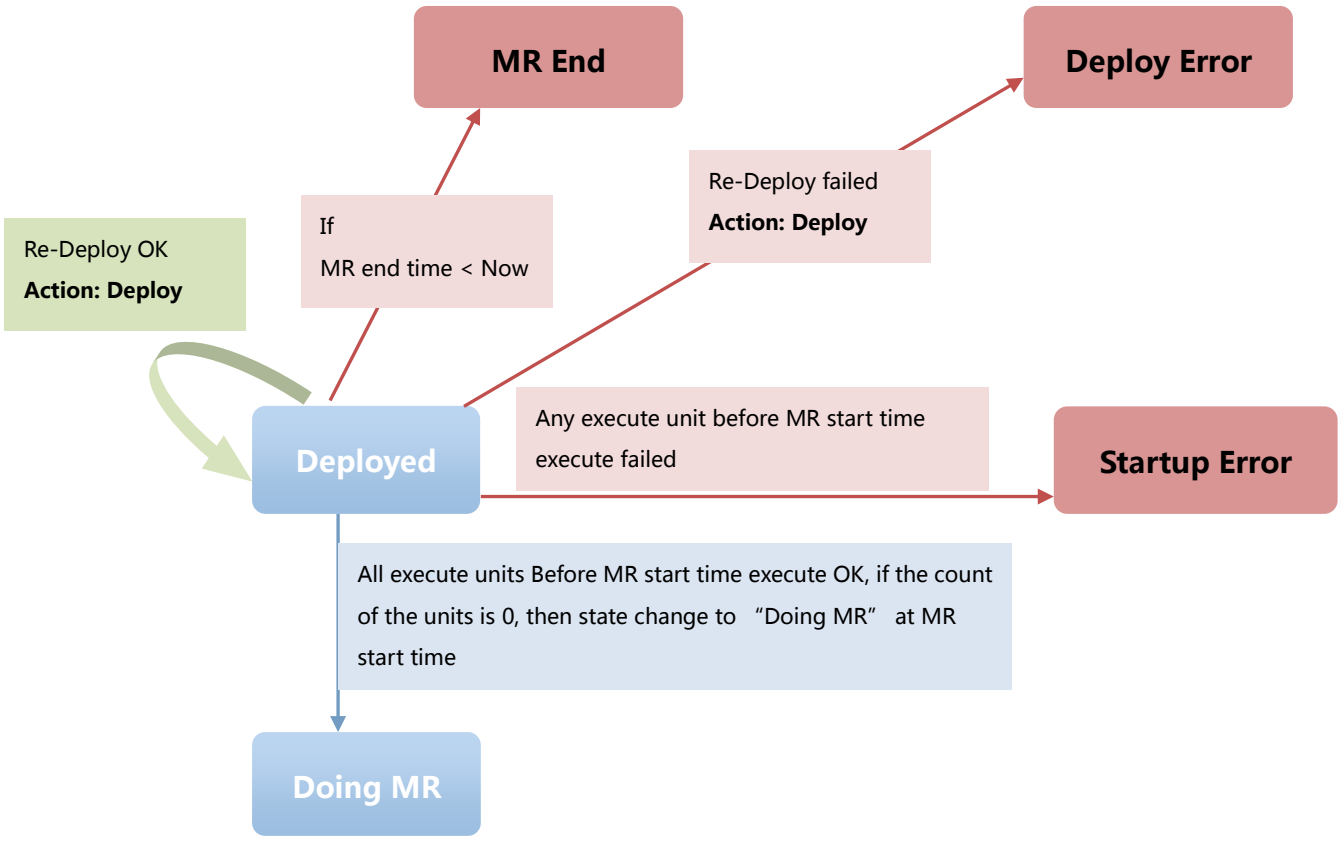


- If Some execute unit: execute time > Now, then those tasks need to manually start by Manually “Restart”, remain execute units would start by schedule



- Else, All the execute units would start by Schedule, client just wait for MR start

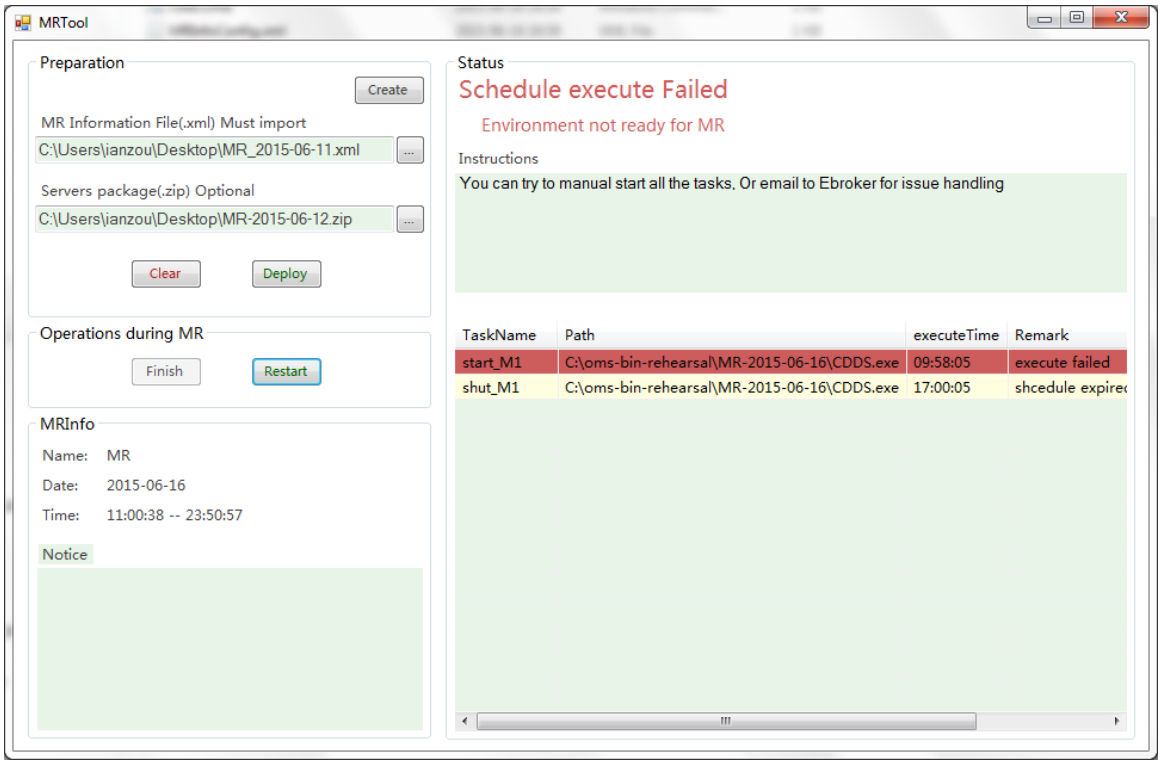


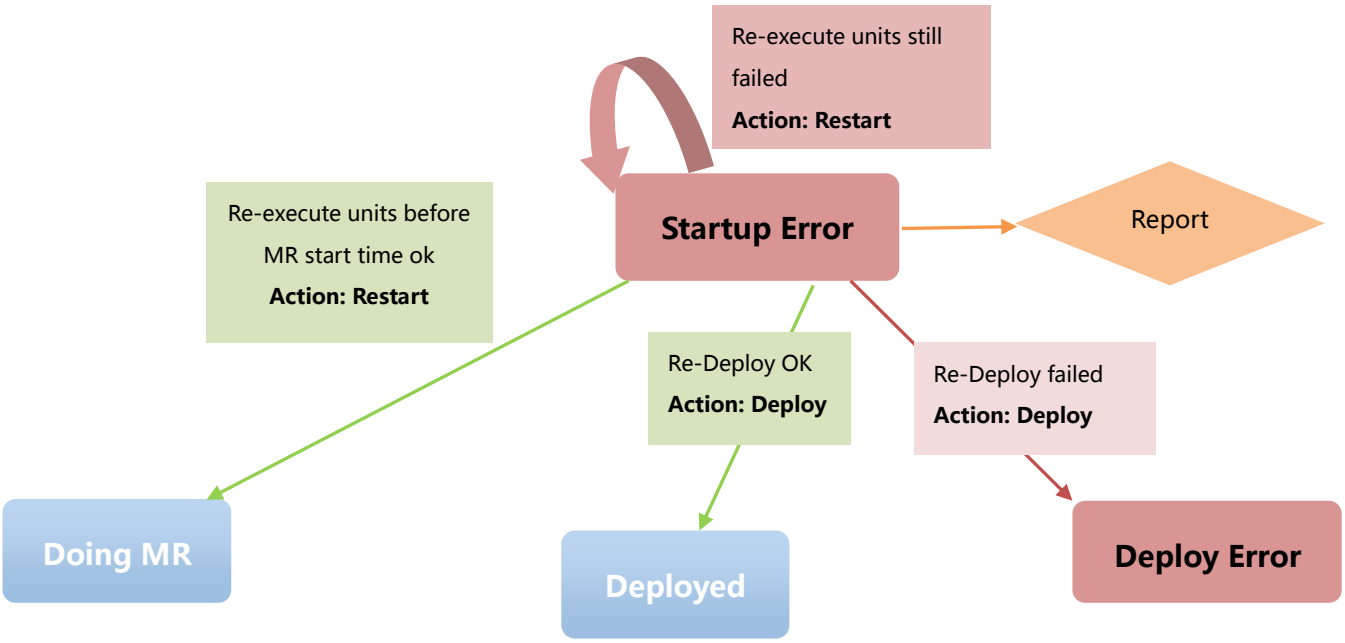


Startup Error State

If any execute units before MR start time execute failed, MR state would turn into "Startup Error" State from "Deployed" state.

In this state, three actions can take: Restart and Deploy, Clear.

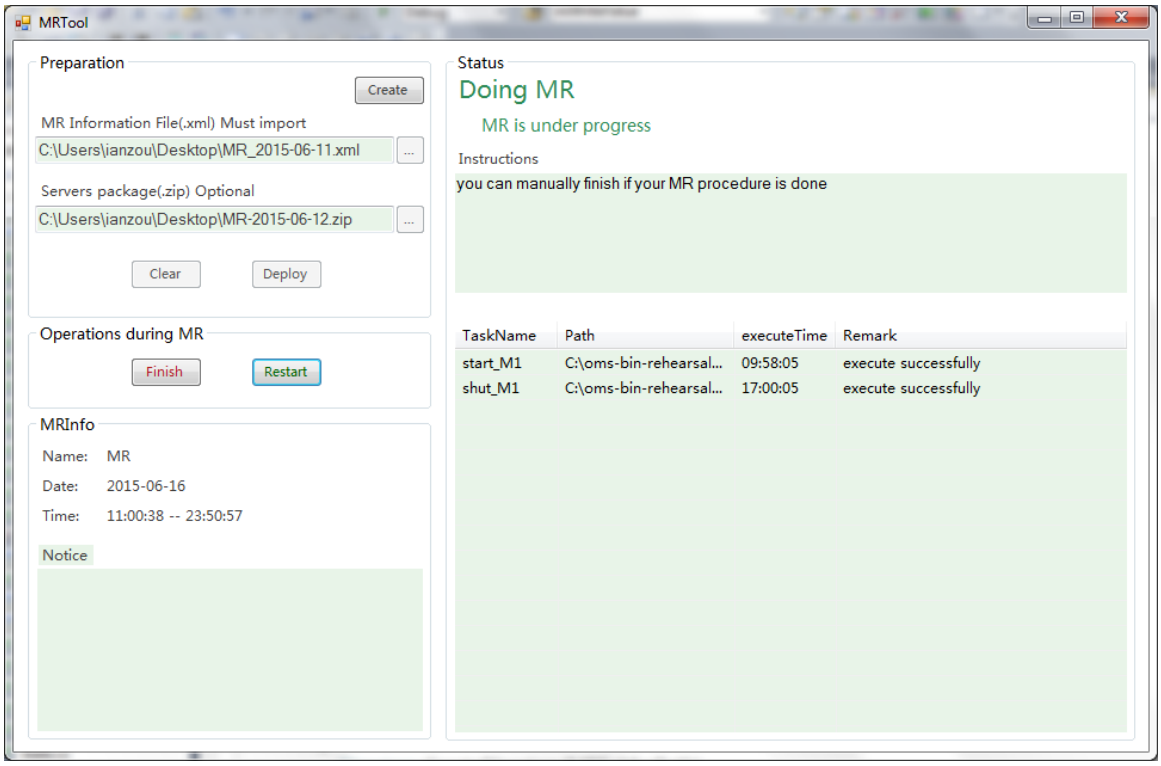


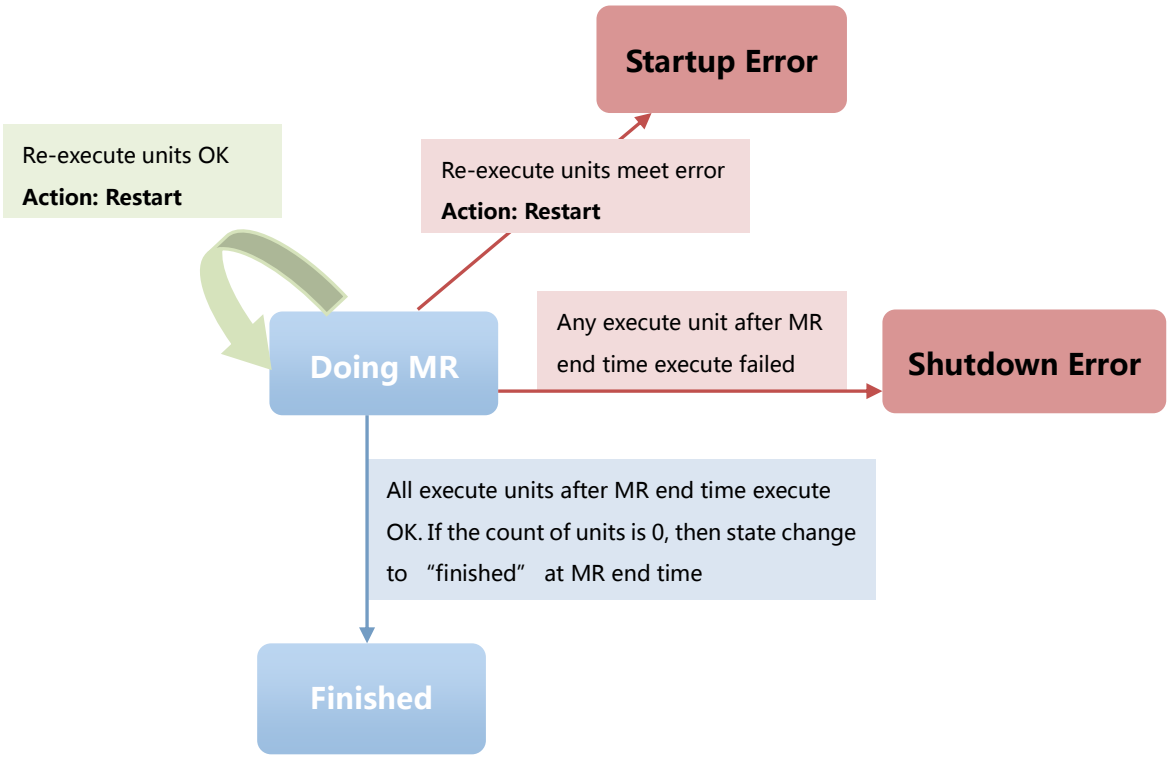


Doing MR state

In this state, two actions can be done: Restart and Finish

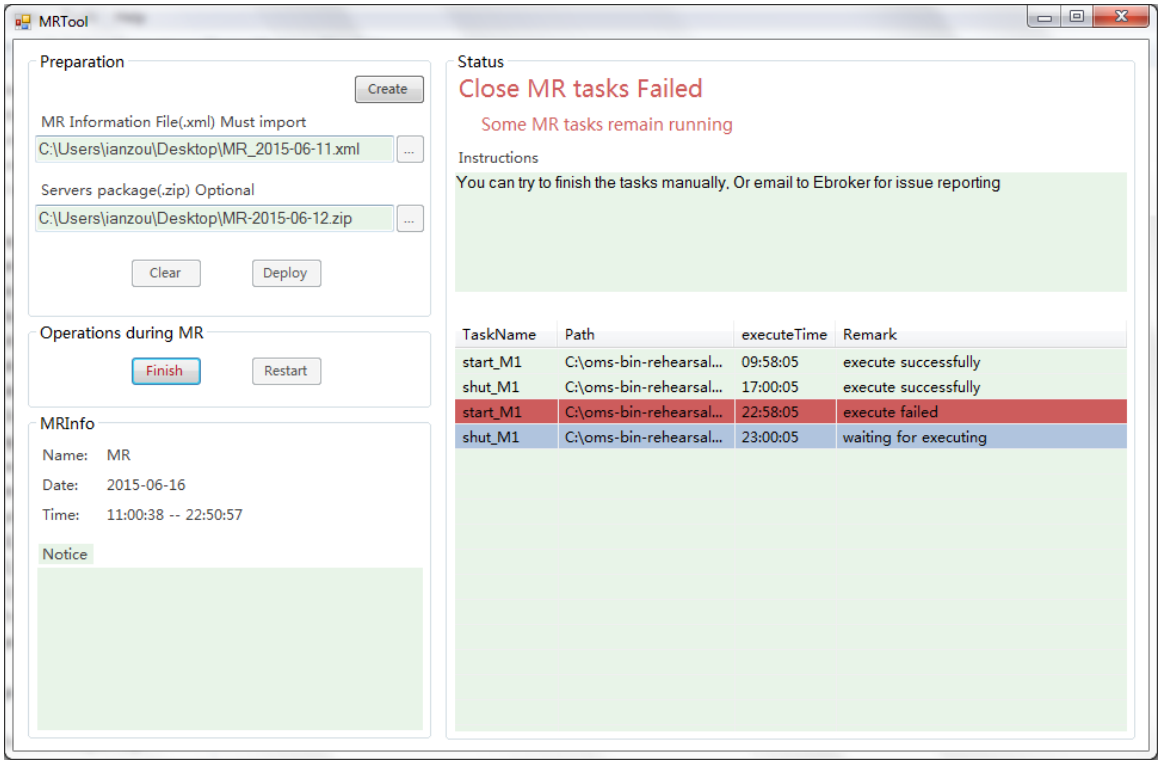
- Issue scenario: When Client meets issues during MR and those issues seem to be due to Programs themselves, which out of the control of MR tool, then client can choose to Restart.
- Client can manually click Finish to execute those execute units which after MR end time **in advance**

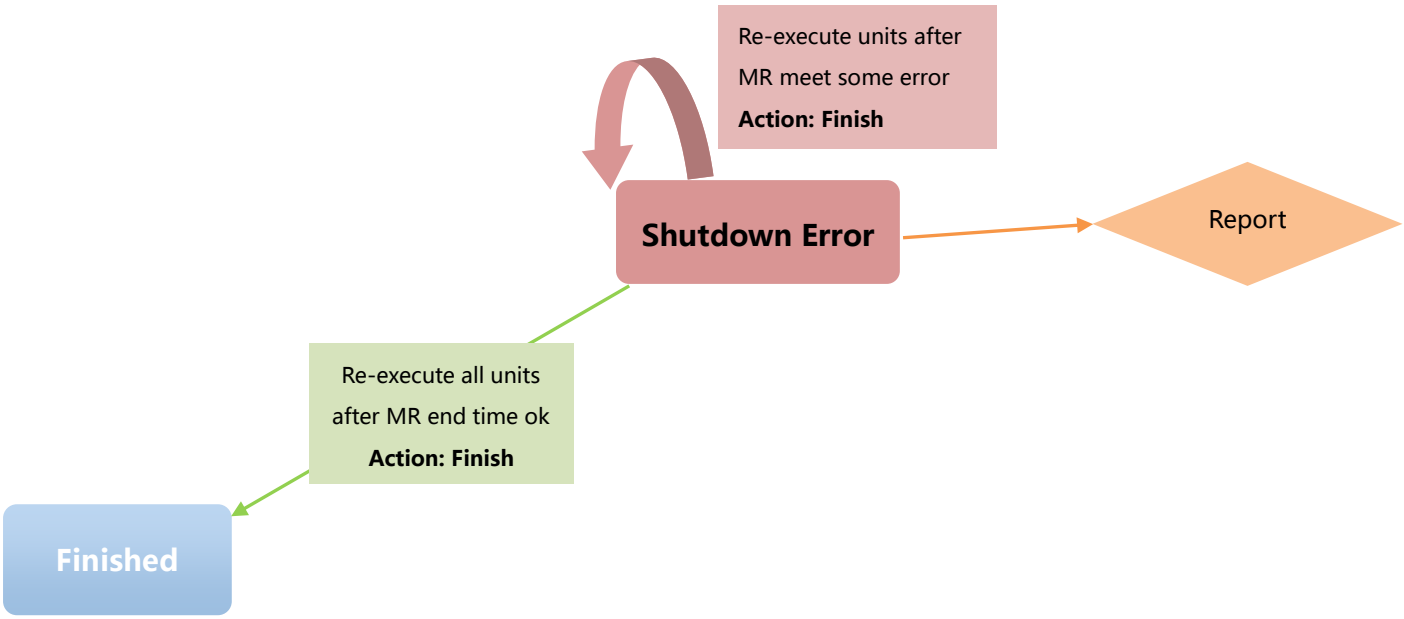




Finish Error state

If any execute unit whose execute time is after MR (like shut down program or Settle jobs) executes failed, state would turn into Finish Error state from Doing MR state. In this state, only one action available: Finish.



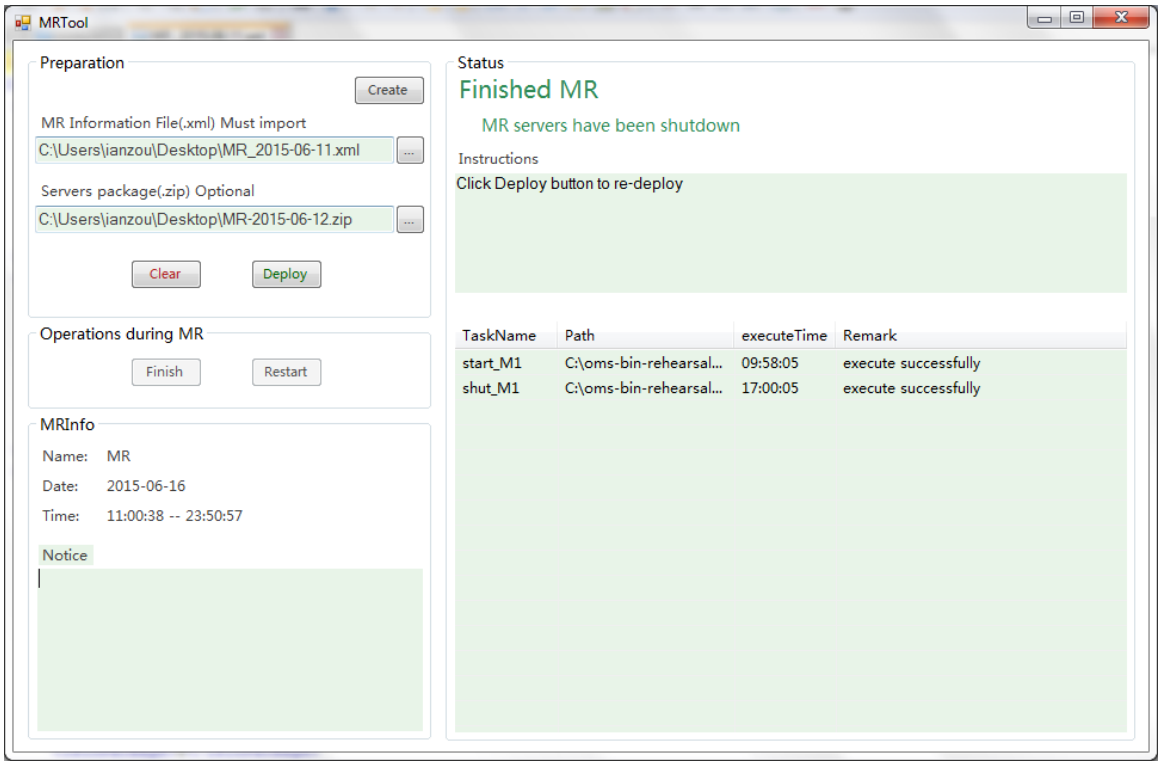


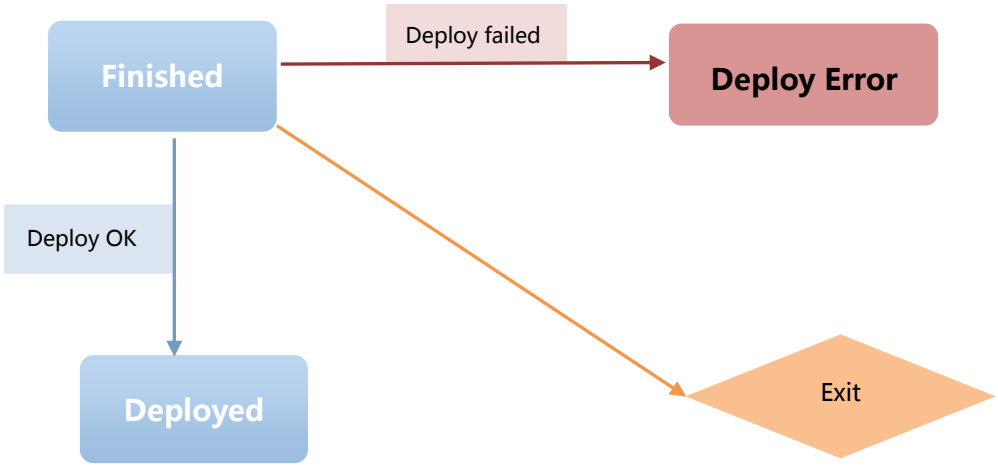
Finished state

If all the execute units whose execute time is after MR (like shut down program or Settle jobs) execute successfully, state would turn into Finished state from Doing MR state.

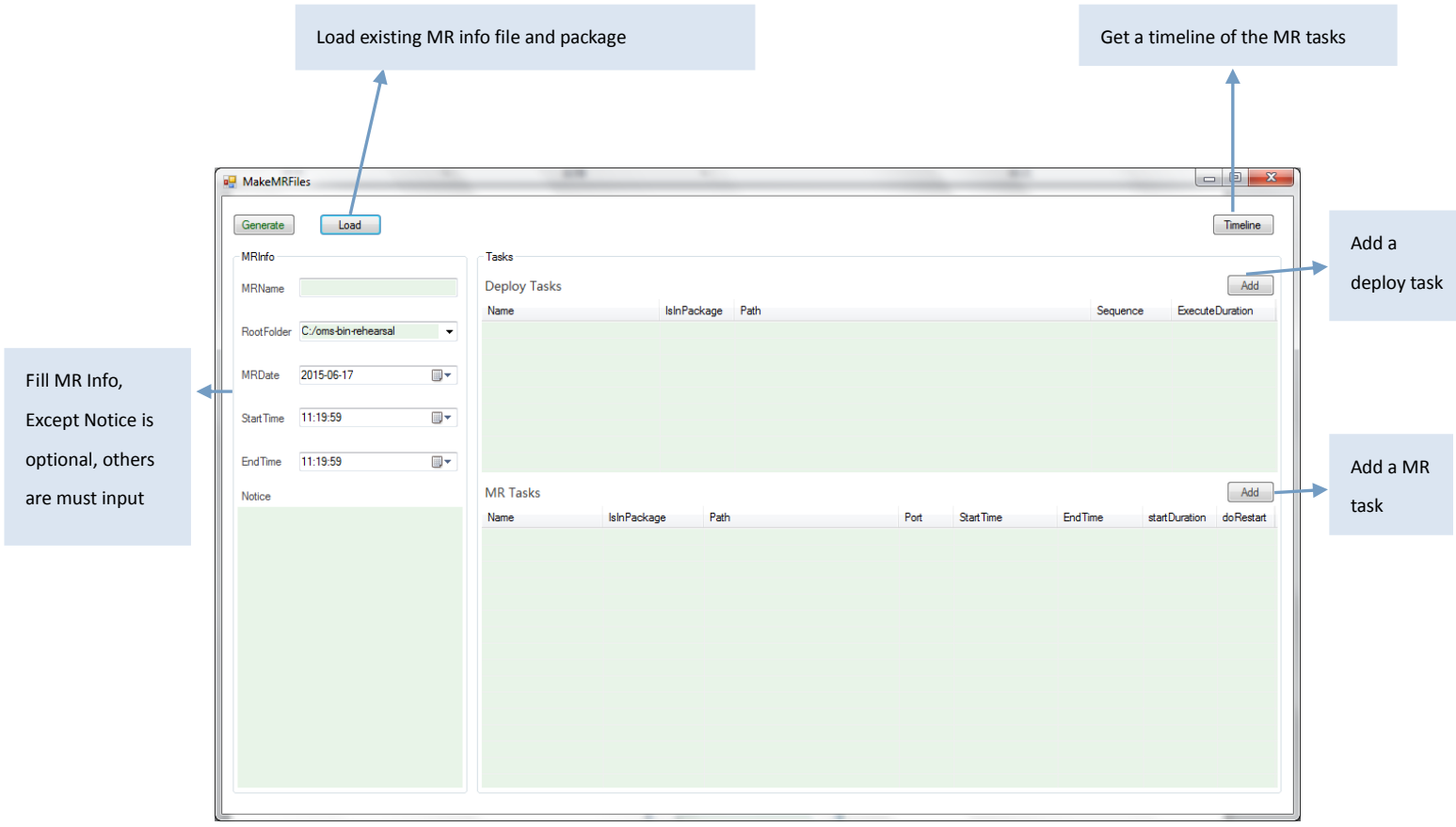
In this state, two actions available: Deploy and Clear

- Client can choose to re-deploy, else, the MR is finished, and client can close the program. Also the environment has resumed

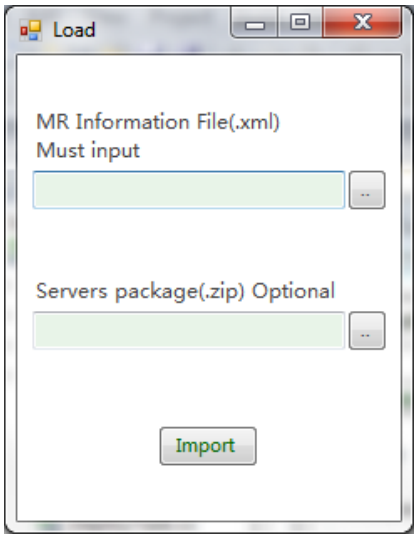




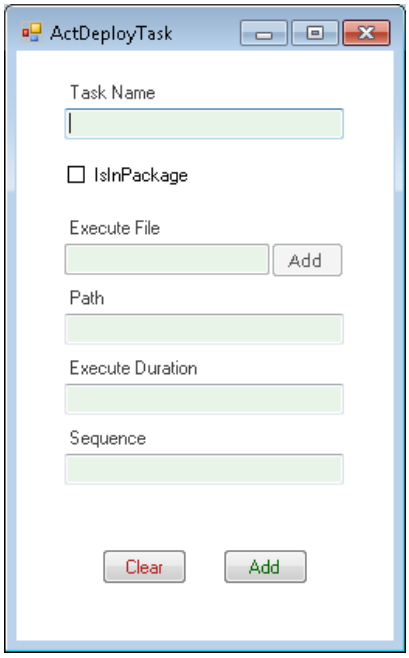
Generate MR Info File and Package



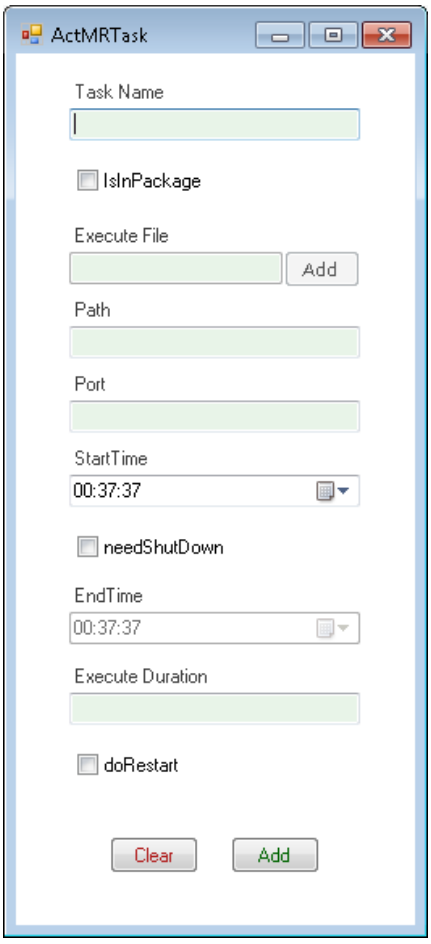
- Load existing MR Info file and Package



- Add a deploy task



- Add a MR task



- Click Timeline button to view the timeline of MR tasks

- Then click the generate button, the MR Info file and Package would be generate to the folder of MRTool