

Using RunAJob for Windows environment. (Developed in C#)

Description:

This Console program is used to schedule Informatica Cloud task from windows environment, is currently in Beta, and under heavy improvements.

Informatica Cloud provide and REST API v2.0 which is used for remote call/response for many types of calls. For this is being used to remote schedule task, and check if the task actually run or not.

Setup and Configuration:

- Download the and unzip file into your preferred directory. (C:\Temp)
- In the directory you will found 3 files
 - RunAJob.exe : Is the console executable which actually connect to ICS
 - InitialProperties.xml : XML which contain the initial properties.
 - bat.bat : A bat file example.
- Provide your credential and server url information in InitialProperties.xml

```
<?xml version="1.0" encoding="utf-8"?>
<root>
  <username>mylogin@ic.com</username>
  <password>mypassword</password>
  <baseurl>https://app.informaticaondemand.com</baseurl>
  <port></port>
  <doDebug>true</doDebug>
</root>
```

<doDebug> is being used to return a simple debug information about where it being in the process to schedule a task. Example output:

```
9/5/2013 4:14:00 PM LogIn      => Logged into Informatica Cloud Services Using REST API 2.0, now scheduling the job
9/5/2013 4:14:01 PM Job       => Task Scheduled [REDACTED], Now checking Activity Monitor
9/5/2013 4:14:03 PM ActivityMonitor => [REDACTED] Still running .....
9/5/2013 4:14:10 PM ActivityLog => Success success rows:15 failed rows:0 start at:16:13:56 end at:16:14:02 took:6 seconds
```

- This utility runs a given task (job) on Informatica Cloud. It requires the following information:
example:
RunAJob <TaskId> <TaskName> <TaskType> <[doActivityCheck]>
where:

- **<Task Id>**: This is the ID of the task as it shows up on the URL when you open the task on the UI. It's the last alphanumeric value in the URL.

The screenshot shows the Informatica Cloud web interface. The browser address bar displays the URL: <https://app.informaticaondemand.com/saas/app/dssTask.do?wzAction=loadExisting&id=0006K40I00000000000W>. The page title is "View Data Synchronization Task". Below the title, it says "This page displays data synchronization task details. Click Edit to configure the task." There are "Edit" and "Done" buttons. The "Task Details" section includes:

Task Name:	Lab_Incremental
Description:	sample to show implementing incremental processing using \$LastRunTime in a filter condition
Operation:	Insert
Wizard Mode:	Basic
Created On:	Oct 13, 2010 11:07:39 PM
Updated On:	Jan 13, 2012 11:11:10 AM
Created By:	eshaikh@informatica.com
Updated By:	eshaikh@informatica.com
Last Run:	Nov 2, 2012 9:03:17 AM

The "Source Details" section includes:

Connection:	SFDC_Dev_es
Source Objects:	Account

The "Target Details" section includes:

Connection:	del_M_gen_Trg
Target Object:	inc_Account.csv

- **<TaskName>** : Used for check the task against ActivityMonitor and ActivityLog API.
- **<TaskType>** : could be DSS or Workflow, depending the task type.
- **[doActivityCheck] = <true/false>**, in case you want to check and wait for the task run.
If you just want to fire and drop set up in false.
If you want to wait set up in true.

- Open and modify the bat.bat file

```
RunAJob 0006D40I000000000000IF My_Task_tsk DSS true
if %ERRORLEVEL% EQU 1 RunAJob 0006D40I000000000000IF My_Task_tsk DSS true
if %ERRORLEVEL% EQU 3 RunAJob 0006D40I000000000000IF My_Task_tsk DSS true
if %ERRORLEVEL% EQU 2 RunAJob 0006D40I000000000000IF My_Task_tsk DSS true
if %ERRORLEVEL% EQU 999 echo "Failed at load InitialProperties.xml"
```

- Return Value (%ERRORLEVEL%) - The utility returns an integer value indicating the status of the run:

Value	Status
1	Success
2	With Error
3	Failed
999	Other Error.

Special Features:

- It retry 10 times to log into ICS, then it will stop after 50 retries with error 999.
- It retry 10 times to schedule a task, then it will stop after 50 retries with error 999
- If the task has been scheduled and could not confirm in ActivityMonitor it retry for ActivityMonitor 100 times (remote calls), then stop and will check in ActivityLog.
- When is checking in activity log, it get the first 50 entries, so if many task run at the same time, and it get the entry 51, then it will stop after 50 retries with error 999