# Repox install and configuration documentation

## Installing the server

### Obtain UNIX the Java install file

Download the latest UNIX installer from <http://sysresearch.org/repox/download.html>. The version used in this install was “REPOX v2.3.7 Linux (2013-10-30)”. Download and unpack this file on the intended server.

### Run the Java file

Unlike the Windows version, the UNIX installer is a shell script and does not have a GUI interface. In addition, the install file seems to be bugged and will require manual intervention to complete the install process correctly.

Sample run…

root:/usr/local/repoxunix#

root:/usr/local/repoxunix# install.sh

REPOX URN syntax is urn:[NAME]: where [NAME] should an identifier of this repox like the dns of the machine. Ex: urn:repox.ist.utl.pt:

Insert REPOX URN [urn:repox.ist.utl.pt:]: urn:[*insert your urn here]*:

Insert Jetty Server port [8080]: [*8080 for example*]

Insert data path [/opt/repoxdata]: /opt/repoxdata

Insert Receiver Admin email: [*insert appropriate email*]

Enter the admin email password:

Enter valid SMTP email server (empty for no admin emails): [*insert SMTP email server*]

Enter the SMTP Port: [*insert SMTP Port like 25*]

Sender email: [*insert sender email*]

Insert User Login: [*insert user login*]

Enter Password: [*insert password for user*] and confirm

java.io.FileNotFoundException: ./jetty/webapps/repox/resources/data/users.xml (No such file or directory)

at java.io.FileOutputStream.open(Native Method)

at java.io.FileOutputStream.<init>(FileOutputStream.java:221)

at java.io.FileOutputStream.<init>(FileOutputStream.java:171)

at PasswordGenerator.writePrettyPrint(PasswordGenerator.java:45)

at PasswordGenerator.main(PasswordGenerator.java:37)

ln: creating symbolic link `/etc/rc0.d/K20jetty': File exists

ln: creating symbolic link `/etc/rc1.d/K20jetty': File exists

ln: creating symbolic link `/etc/rc2.d/S20jetty': File exists

ln: creating symbolic link `/etc/rc3.d/S20jetty': File exists

ln: creating symbolic link `/etc/rc4.d/S20jetty': File exists

ln: creating symbolic link `/etc/rc5.d/S20jetty': File exists

ln: creating symbolic link `/etc/rc6.d/K20jetty': File exists

Found JAVA=/usr/lib/jvm/java-1.7.0-openjdk-1.7.0.79.x86\_64/jre/bin/java in JAVA\_HOME=/

Starting Jetty: STARTED Jetty Mon May 11 11:49:10 EDT 2015

root:/usr/local/repoxunix# 2015-05-11 10:49:10.527::INFO: Logging to STDERR via org.mortbay.log.StdErrLog

2015-05-11 10:49:10.604::INFO: Redirecting stderr/stdout to /usr/local/repoxunix/jetty/logs/2015\_05\_11.stderrout.log

root:/usr/local/repoxunix#

root:/usr/local/repoxunix#

At this point the installer has finished, and the service has started. However, because the user account information has not been copied to the correct file (the “java.io.FileNotFoundException” errors), you will find yourself unable to log in. To correct this, first stop the Repox service.

root:/usr/local/repoxunix# /usr/local/repoxunix/jetty/bin/jetty.sh stop

Found JAVA=/usr/lib/jvm/java-1.7.0-openjdk-1.7.0.79.x86\_64/jre/bin/java in JAVA\_HOME=/

Stopping Jetty: OK

root:/usr/local/repoxunix#

At this point, the required directory structure to write the “users.xml” file is not present. To complete the required directory structure started by the install, use the following command…

root:/usr/local/repoxunix#

root:/usr/local/repoxunix# mkdir /usr/local/repoxunix/jetty/webapps/repox/resources/data

root:/usr/local/repoxunix#

When that is done, run the install script again with the same options as before. This time the install process should complete without error, with the administrator user name and hashed password written to the “users.xml”, though it will still be in the wrong location.

After the install is done a second time, the “users.xml” file will correctly updated with the administrator login information need to be copied to the correct location in the data directory structure. To do this, first stop the Repox server that was started again as part of the second install process.

root:/usr/local/repoxunix# /usr/local/repoxunix/jetty/bin/jetty.sh stop

Found JAVA=/usr/lib/jvm/java-1.7.0-openjdk-1.7.0.79.x86\_64/jre/bin/java in JAVA\_HOME=/

Stopping Jetty: OK

root:/usr/local/repoxunix#

Next copy the “users.xml” file to the correct location.

root:/usr/local/repoxunix# mv /opt/repoxdata/configuration/users.xml /opt/repoxdata/configuration/old.users.xml

root:/usr/local/repoxunix# cp /usr/local/repoxunix/jetty/webapps/repox/resources/data/users.xml /opt/repoxdata/configuration/users.xml

Once the “users.xml” file is copied to correct location in the Repox data directory, start the server.

root:/usr/local/repoxunix#

root:/usr/local/repoxunix# /usr/local/repoxunix/jetty/bin/jetty.sh start

From here, verify that you are able to log into Repox using the expected user name and password using the link below:

<http://urn:port/repox/>

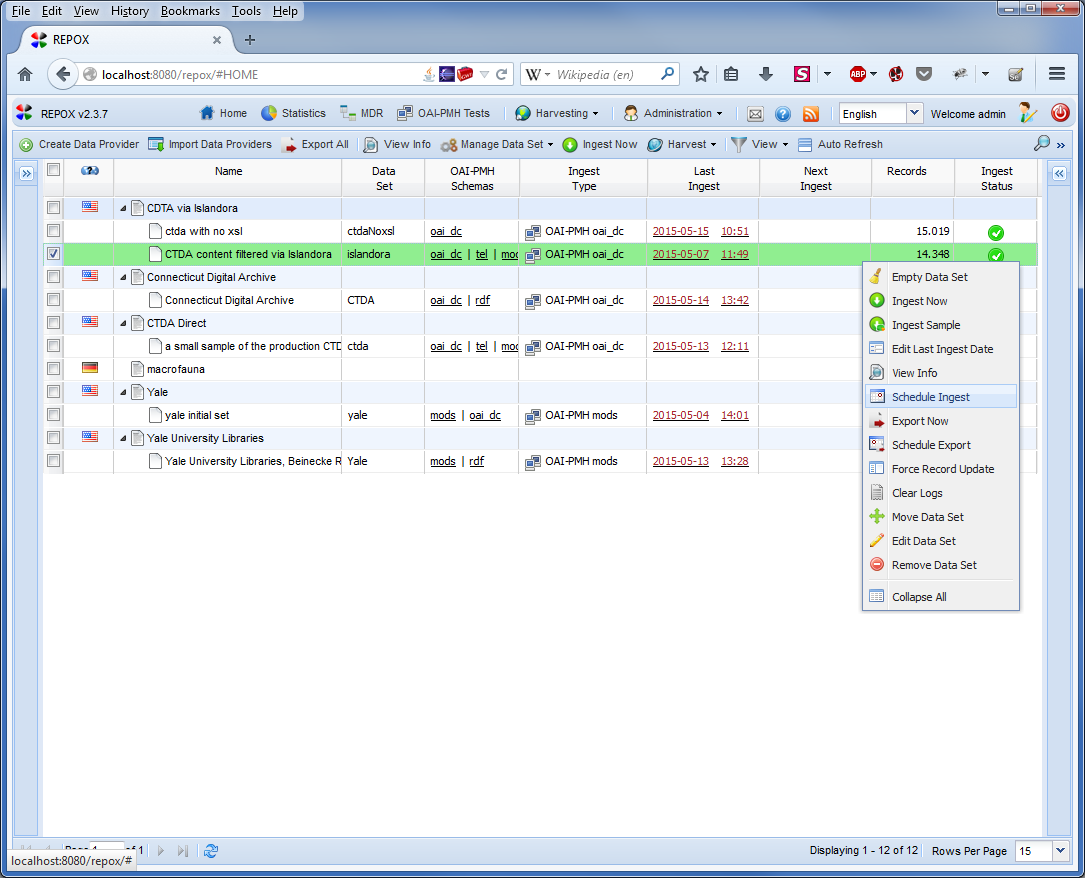
At this point Repox should be ready for use provided access has been enabled from the outside. If not, you would need to open port 8080 on the server to allow access by external OAI harvesters once feeds have been configured. For information on configuring feed sources in Repox please refer to the application documentation.

<http://repox.sysresearch.org/doc.html>

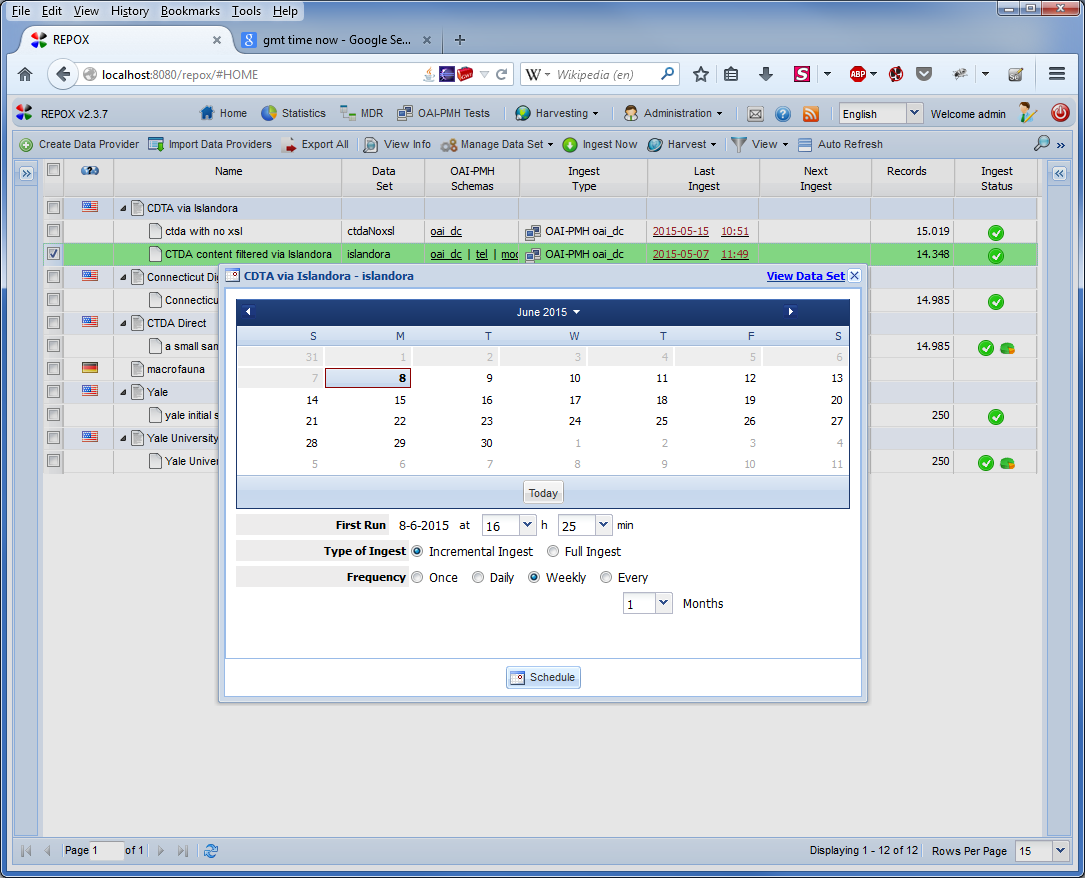
## Scheduling OAI harvesting

Normally, scheduling regular data retrievals from OAI harvesting from data sources is done via the Repox user interface. While this works for setting up initial harvesting configurations, viewing and editing existing scheduled harvesting configurations appears to be bugged in the application. Because of this, manually editing (or viewing) of the scheduled tasks files may be required.

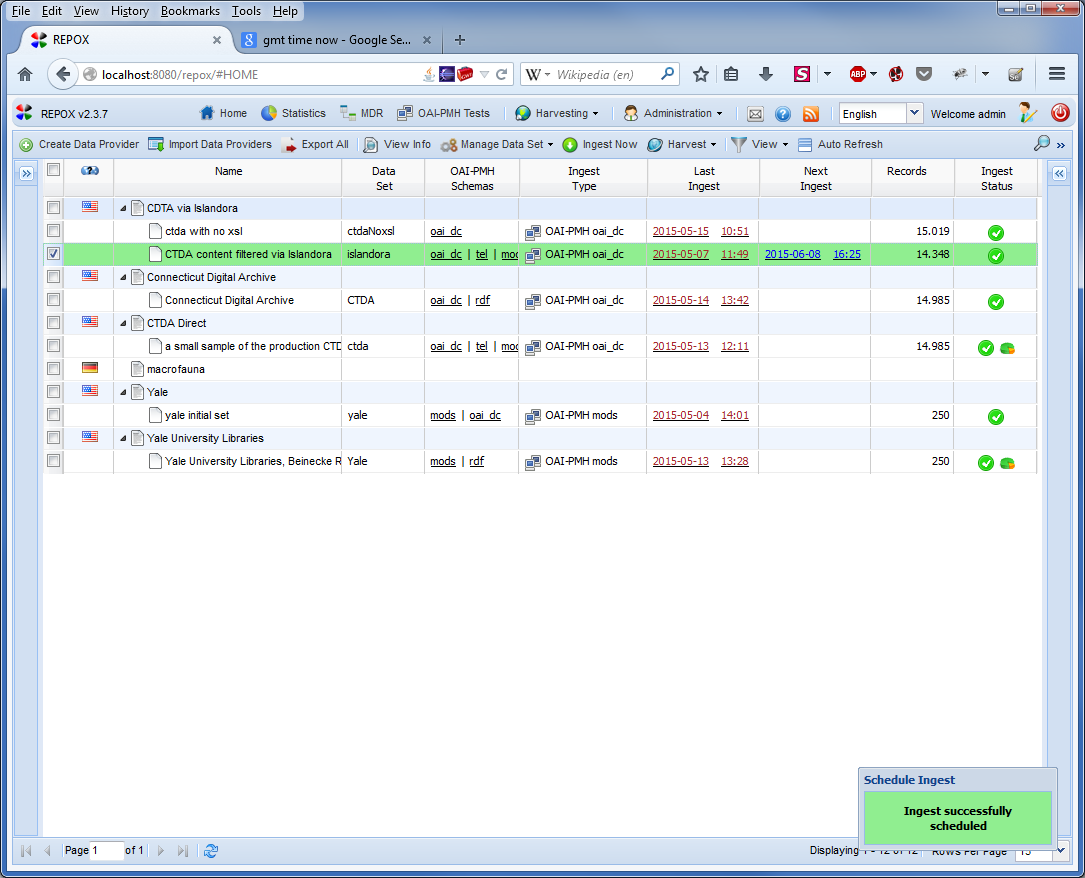
To schedule a task, first select a configured data provider. In this case, the “CTDA content filtered via Islandora” provider has been selected. Next, right-mouse click on the prorvider and then select the “Schedule Ingest” from the context menu.



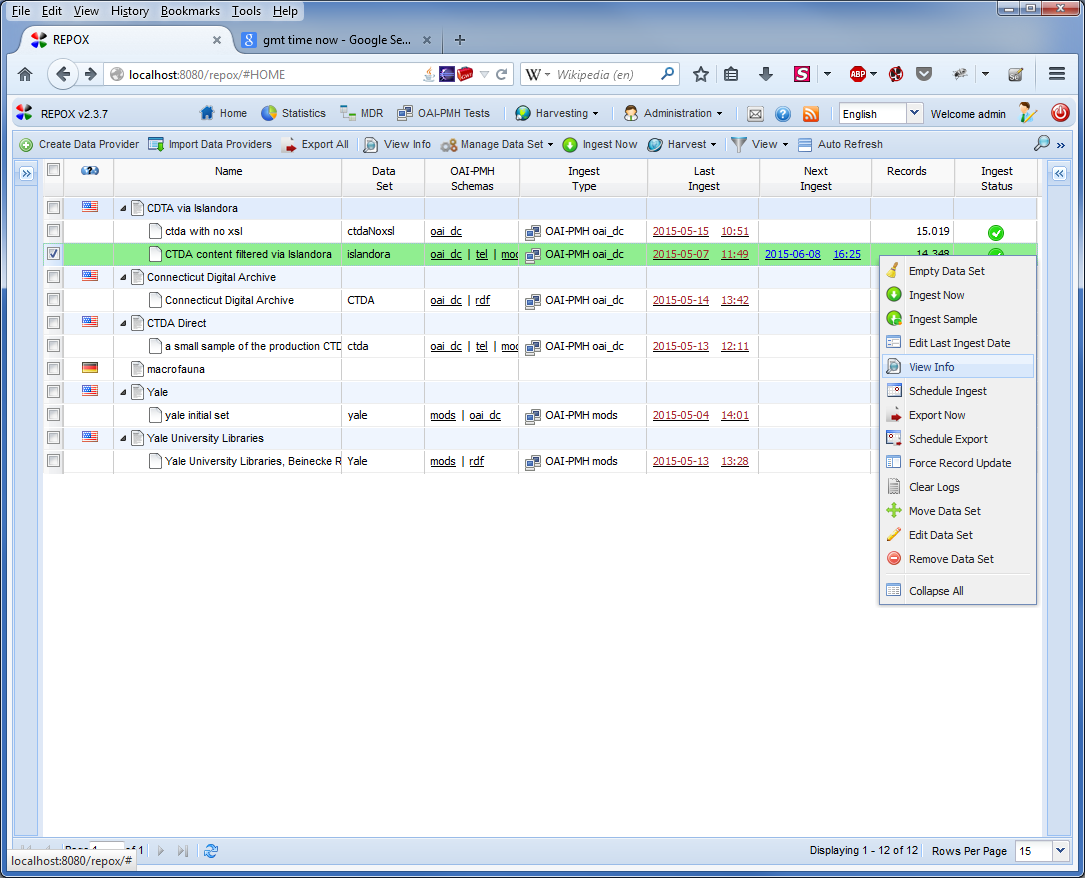
From the schedule dialog box, select the date and parameters that you wish for the scheduled task from the list of options. Here, the task has been configured to perform a weekly incremental ingest starting on June 8 at 4:25 PM. Is should be noted that all times appear to be in GMT even though this is not mentioned.



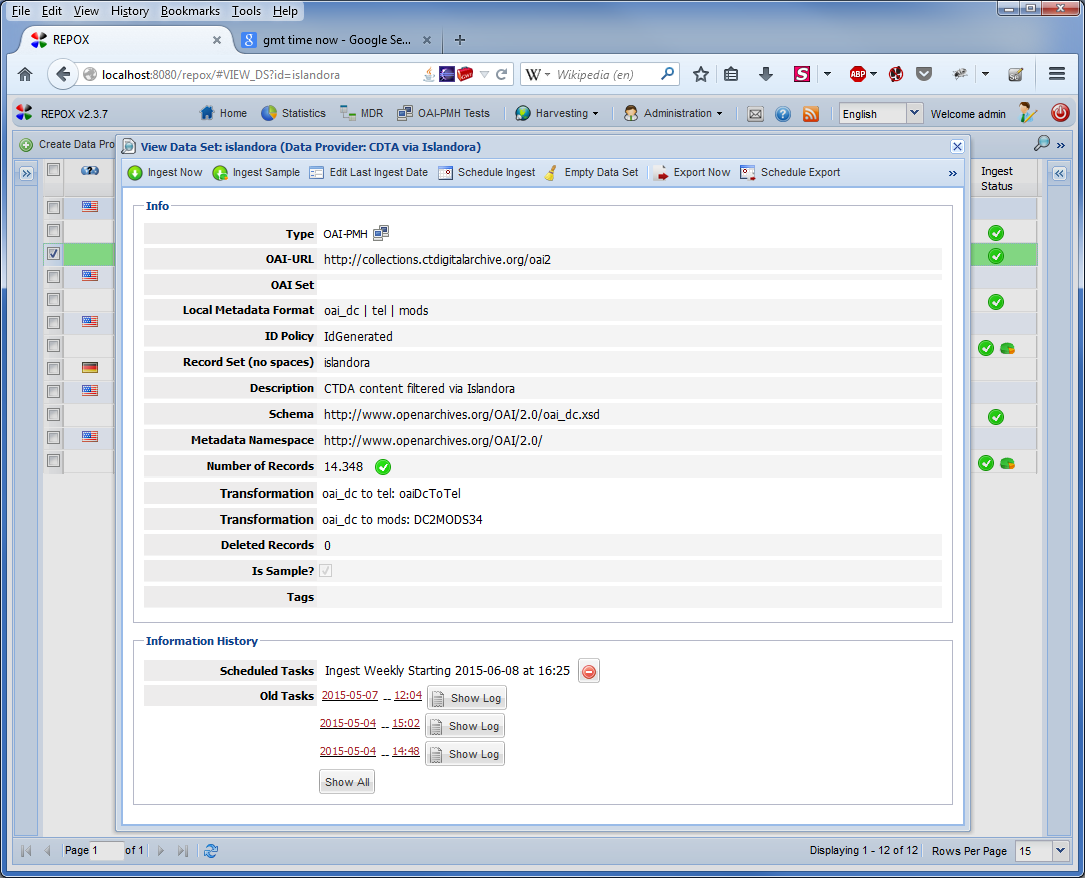
Once the ingest task is scheduled, a message should appear indicating that the task has been added to the schedule successfully. Additionally, the data and time of the next ingest should appear in the provider grid view.



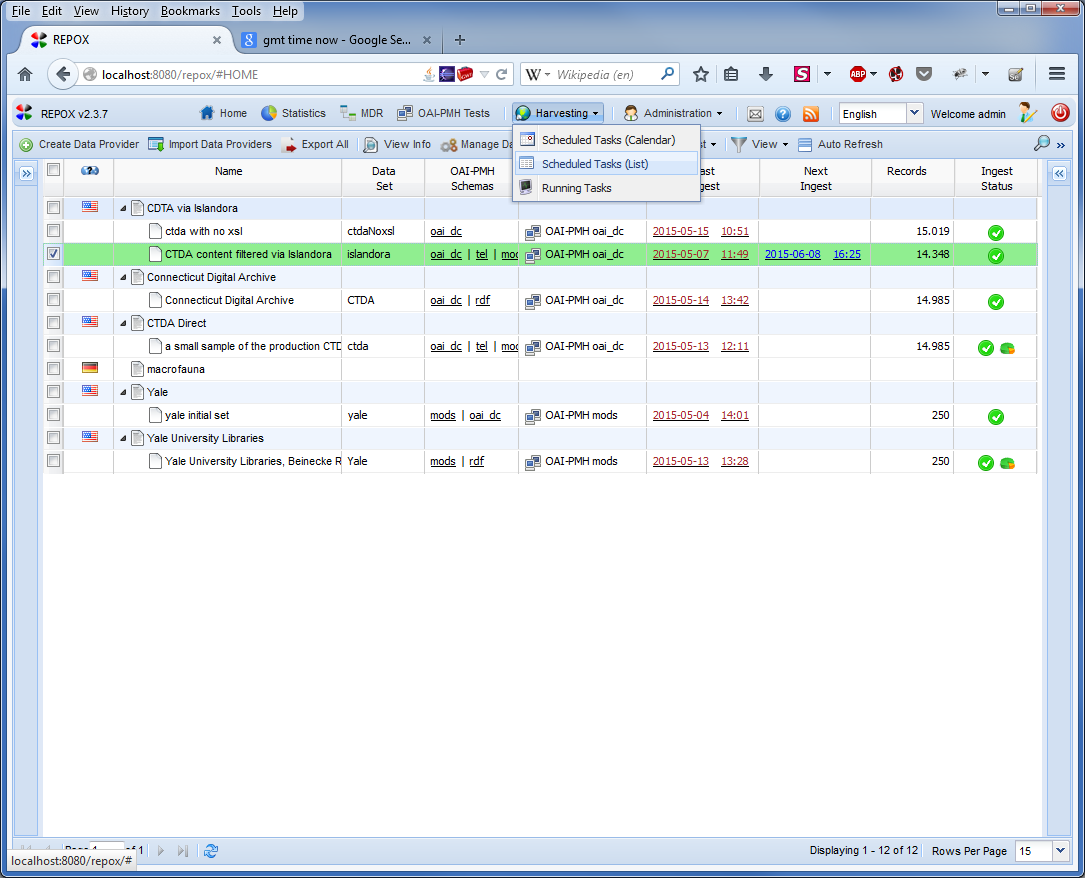
Schedule tasks for a specific provider can also be viewed by selecting a provider, then right mouse clicking and selecting the “View Info” menu item.



This dialog box shows the current status of a provider as well as providing links to any provider related ingest and exports. If a scheduled task for a provider needs to be edited it can be done here.



It should be noted that while other methods to view scheduled tasks appear to be provided in Repox, these methods do not seem to work as expected and omit scheduled tasks. An example of this can be seen by selecting the Harvesting > Scheduled Task (List) menu option.



From here, it can be seen that the resulting view that is generated omits the scheduled ingest task that clearly appears on the provider grid view.

