HPCC Systems®

HPCC Preflight and Certification

Boca Raton Documentation Team

LexisNexis®
Risk Solutions

HPCC Preflight and Certification

Boca Raton Documentation Team

We welcome your comments and feedback about this document via email to `<docfeedback@hpccsystems.com>`

Please include **Documentation Feedback** in the subject line and reference the document name, page numbers, and current Version Number in the text of the message.

All names and example data used in this manual are fictitious. Any similarity to actual persons, living or dead, is purely coincidental.

2015 Version 5.4.2-1

Introduction : Certification Approach

This document provides steps you can use to certify your HPCC environment.

Use this procedure when you create a new HPCC system or after you make any changes or upgrades to the system. This will ensure that your system is functioning properly.

You can also use all or some of these procedures on a regularly scheduled basis or before mission critical data processing.

> We suggest reading this document in its entirety before beginning.

Scope

These procedures certify the following functional areas:

- System Machine Readiness
- Data Transfer from Data Refinery to Landing Zone
- Data Transfer from Landing Zone to Data Refinery
- Data Refinery Functions
  - Certify Data Refinery full sort capabilities
  - Certify Data Refinery local sort capabilities
  - Certify Data Refinery local dedup capabilities
  - Certify Data Refinery hash dedup capabilities
  - Certify Data Refinery compress I/O capabilities
  - Certify Data Refinery string search capabilities
  - Certify Data Refinery Engine key build capabilities
  - Certify Data Delivery Engine access to indexed data
  - Certify Rapid Data Delivery Engine access to indexed data
- hThor Functions
- Thor Functions
- Roxie Functions

Before You Begin

1. Make sure the _*Certification* folder is in your repository. This is typically installed with the IDE.

2. Remove any items that might be left from previous certifications. Search and remove any old Workunits, DFU Workunits, data files, and published queries.

3. Filenames and other variables are defined in the _Certification.Setup file, you can edit this file to change the number of records or filenames produced.

Preflight

The first step in certifying that the platform is installed and configured properly is to run a preflight check on the components. This ensures that all machines are operating and have the proper executables running. This also confirms there is adequate disk space, available memory, and acceptable available CPU % values.

- Open ECL Watch in your browser using the following URL:

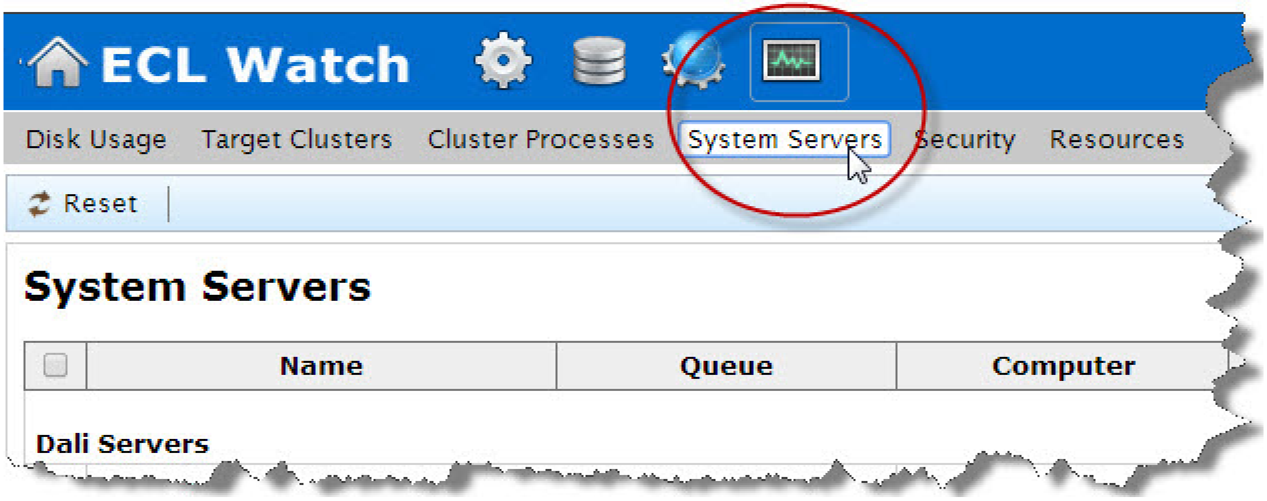
  http://nnn.nnn.nnn.nnn:pppp (where nnn.nnn.nnn.nnn is your ESP Server’s IP Address and pppp is the port. The default port is 8010)

⚠	**Note:** That your IP address could be different from the ones provided in these figures. Please use the IP address provided by your installation.

Preflight System Servers

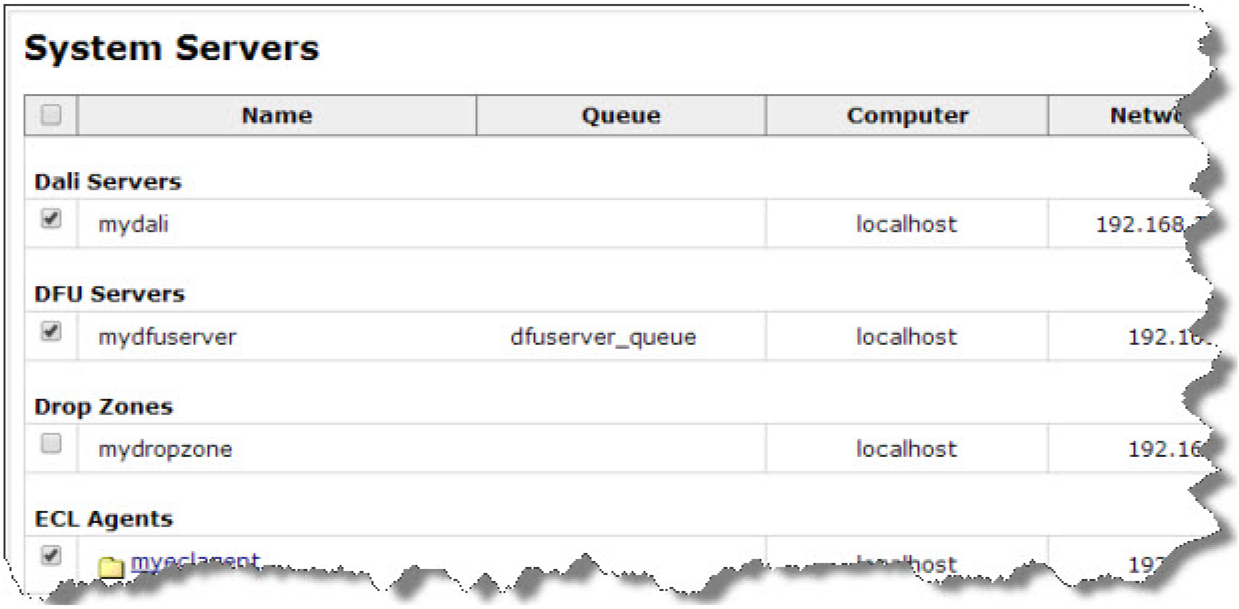
1. Click on the **Operations** icon then click on the **System Servers** link.

   Figure 1. System Servers link



   A screen similar to the following displays.

   Figure 2. System Servers page

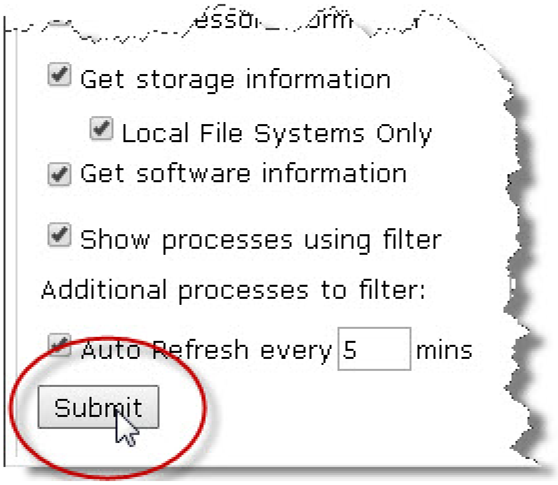


System Servers

☐	Name	Queue	Computer	Netw
Dali Servers				
☑	mydali		localhost	192.168.
DFU Servers				
☑	mydfuserver	dfuserver_queue	localhost	192.16
Drop Zones				
☐	mydropzone		localhost	192.16
ECL Agents				
☑	myeclagent		localhost	19

2. Press the **Submit** button at the bottom of this page to start preflight.

Figure 3. Submit



☑ Get storage information

☑ Local File Systems Only

☑ Get software information

☑ Show processes using filter

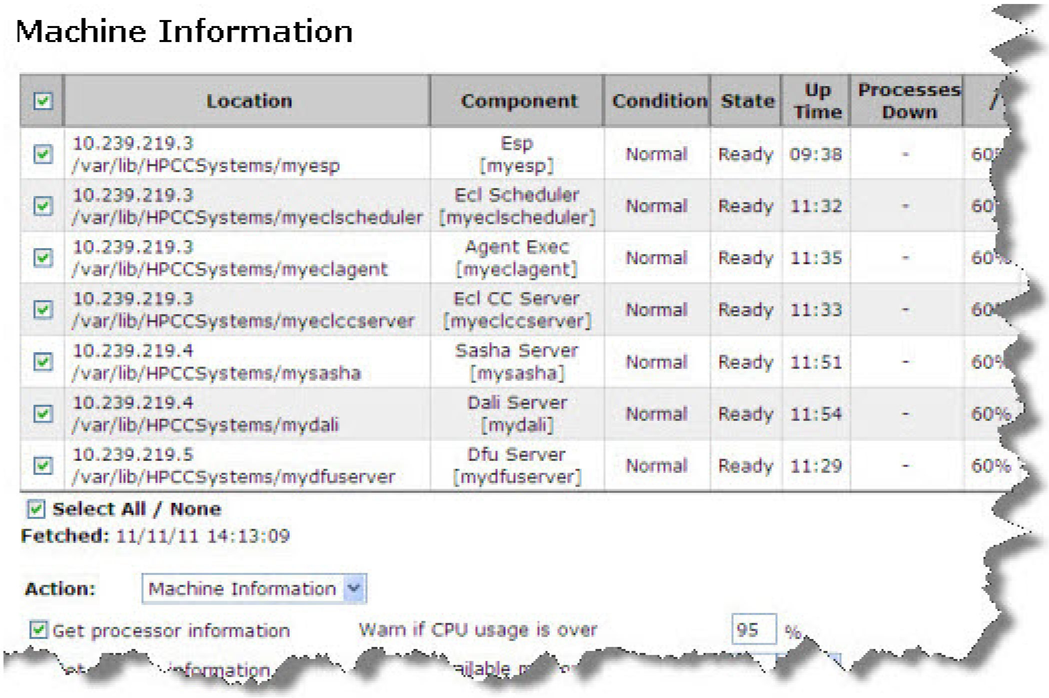
Additional processes to filter:

☑ Auto Refresh every 5 mins

EXPECTED RESULTS:

After pressing Submit, a screen similar to the following displays.

Figure 4. System Component Inforamtion



Machine Information

☑	Location	Component	Condition	State	Up Time	Processes Down	/
☑	10.239.219.3 /var/lib/HPCCSystems/myesp	Esp [myesp]	Normal	Ready	09:38	-	60
☑	10.239.219.3 /var/lib/HPCCSystems/myeclscheduler	Ecl Scheduler [myeclscheduler]	Normal	Ready	11:32	-	60
☑	10.239.219.3 /var/lib/HPCCSystems/myeclagent	Agent Exec [myeclagent]	Normal	Ready	11:35	-	60%
☑	10.239.219.3 /var/lib/HPCCSystems/myeclccserver	Ecl CC Server [myeclccserver]	Normal	Ready	11:33	-	60
☑	10.239.219.4 /var/lib/HPCCSystems/mysasha	Sasha Server [mysasha]	Normal	Ready	11:51	-	60%
☑	10.239.219.4 /var/lib/HPCCSystems/mydali	Dali Server [mydali]	Normal	Ready	11:54	-	60%
☑	10.239.219.5 /var/lib/HPCCSystems/mydfuserver	Dfu Server [mydfuserver]	Normal	Ready	11:29	-	60%

☑ **Select All / None**

Fetched: 11/11/11 14:13:09

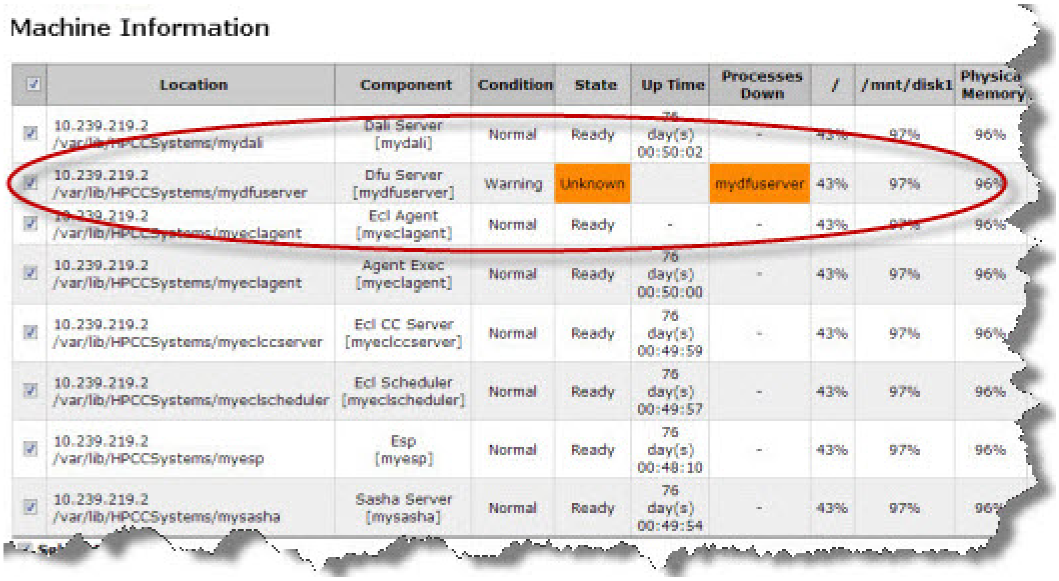
Action: Machine Information

☑ Get processor information   Warn if CPU usage is over 95 %

This screen displays information on several system components. This information indicates whether several components are actually up and running appropriately. The resulting page shows useful information about each component. The component name, the condition, the component state, how long the component has been up and running, the amount of disk usage, memory usage and other information is available at a glance.

If there are any failed components, they are highlighted in orange, indicating they are not ready.

Figure 5. Failed Component



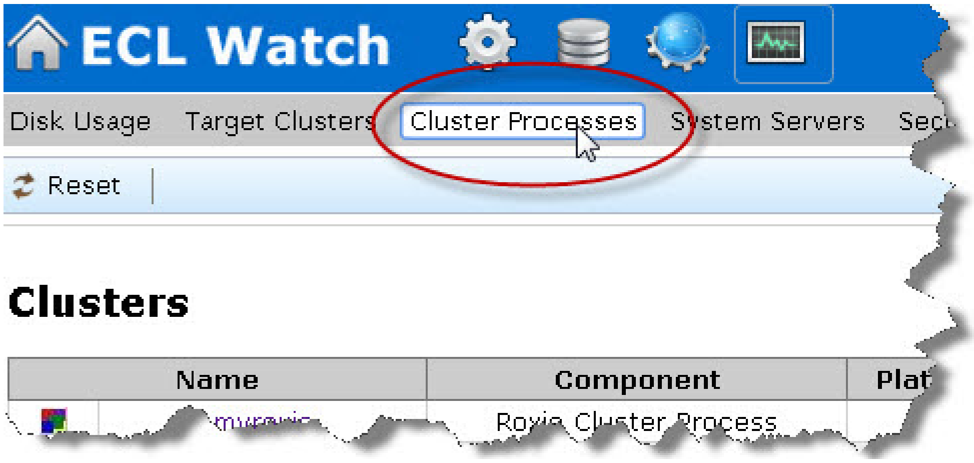
Machine Information

☑	Location	Component	Condition	State	Up Time	Processes Down	/	/mnt/disk1	Physica Memory
☑	10.239.219.2 /var/lib/HPCCSystems/mydali	Dali Server [mydali]	Normal	Ready	76 day(s) 00:50:02	-	43%	97%	96%
☑	10.239.219.2 /var/lib/HPCCSystems/mydfuserver	Dfu Server [mydfuserver]	Warning	Unknown		mydfuserver	43%	97%	96%
☑	10.239.219.2 /var/lib/HPCCSystems/myeclagent	Ecl Agent [myeclagent]	Normal	Ready	-	-	43%	97%	96%
☑	10.239.219.2 /var/lib/HPCCSystems/myeclagent	Agent Exec [myeclagent]	Normal	Ready	76 day(s) 00:50:00	-	43%	97%	96%
☑	10.239.219.2 /var/lib/HPCCSystems/myeclccserver	Ecl CC Server [myeclccserver]	Normal	Ready	76 day(s) 00:49:59	-	43%	97%	96%
☑	10.239.219.2 /var/lib/HPCCSystems/myeclscheduler	Ecl Scheduler [myeclscheduler]	Normal	Ready	76 day(s) 00:49:57	-	43%	97%	96%
☑	10.239.219.2 /var/lib/HPCCSystems/myesp	Esp [myesp]	Normal	Ready	76 day(s) 00:48:10	-	43%	97%	96%
☑	10.239.219.2 /var/lib/HPCCSystems/mysasha	Sasha Server [mysasha]	Normal	Ready	76 day(s) 00:49:54	-	43%	97%	96%

Preflight Thor

1. Click on the **Operations** icon then click on the **Cluster Processes** link.

Figure 6. Cluster Processes Link

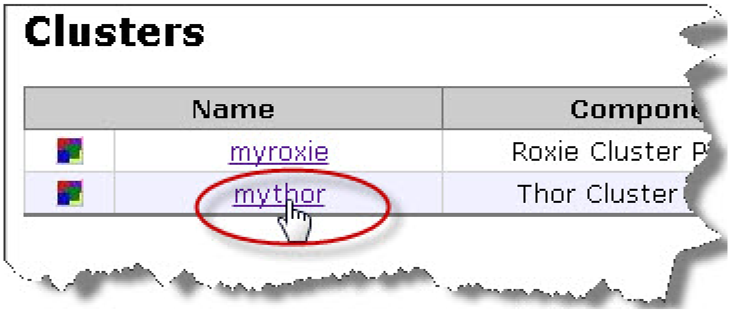


Clusters

Name	Component	Plat
myroxie	Roxie Cluster Process	

2. Click on the **mythor** link.

Figure 7. mythor link



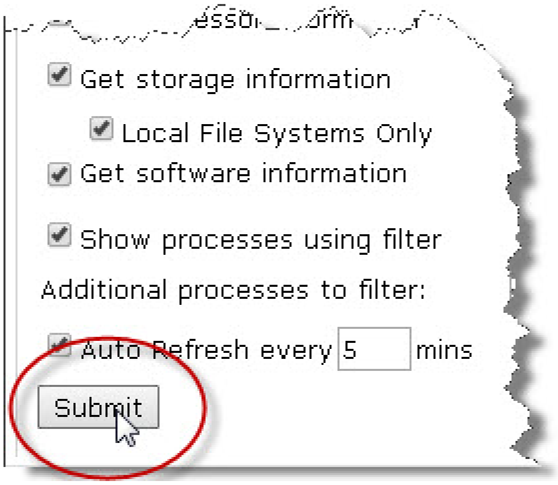
Clusters

	Name	Compone
🟥	myroxie	Roxie Cluster P
🟥	mythor	Thor Cluster

3. Check the **Select All** checkbox (if necessary).

4. Press the **Submit** button to start preflight.

Figure 8. Submit



☑ Get storage information

☑ Local File Systems Only

☑ Get software information

☑ Show processes using filter

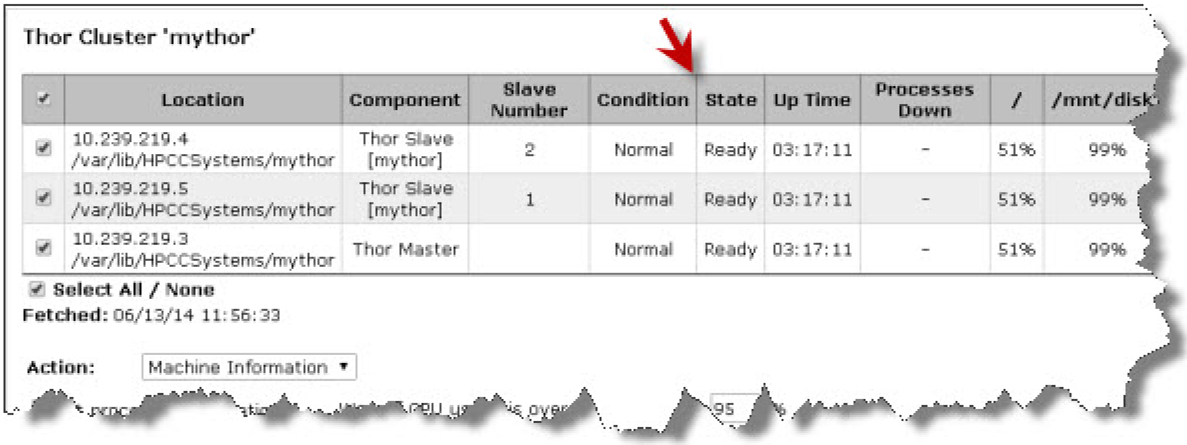
Additional processes to filter:

☑ Auto Refresh every 5 mins

EXPECTED RESULTS:

After pressing Submit, a screen similar to the following should display.

Figure 9. ESP mythor system component information



Thor Cluster 'mythor'

☑	Location	Component	Slave Number	Condition	State	Up Time	Processes Down	/	/mnt/disk
☑	10.239.219.4 /var/lib/HPCCSystems/mythor	Thor Slave [mythor]	2	Normal	Ready	03:17:11	-	51%	99%
☑	10.239.219.5 /var/lib/HPCCSystems/mythor	Thor Slave [mythor]	1	Normal	Ready	03:17:11	-	51%	99%
☑	10.239.219.3 /var/lib/HPCCSystems/mythor	Thor Master		Normal	Ready	03:17:11	-	51%	99%

☑ Select All / None

Fetched: 06/13/14 11:56:33

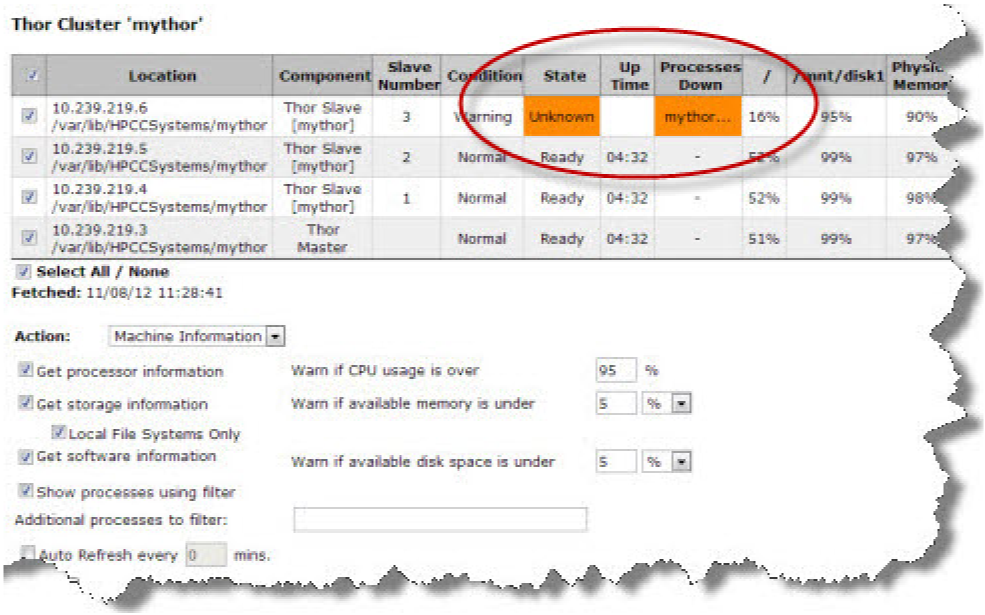
Action: Machine Information ▾

This screen displays information on Thor components. This information indicates whether the components are actually up and running appropriately. The resulting page shows useful information about each component. The component name, the condition, the component state, how long the component has been up and running, the amount of disk usage, memory usage and other information is available at a glance.

	If your system has more than 1 Thor cluster, repeat these steps for each cluster.

If there are any failed components, they are highlighted in orange, indicating they are not ready.

Figure 10. Failed Component



Thor Cluster 'mythor'

☑	Location	Component	Slave Number	Condition	State	Up Time	Processes Down	/	/mnt/disk1	Physic Memor
☑	10.239.219.6 /var/lib/HPCCSystems/mythor	Thor Slave [mythor]	3	Warning	Unknown		mythor...	16%	95%	90%
☑	10.239.219.5 /var/lib/HPCCSystems/mythor	Thor Slave [mythor]	2	Normal	Ready	04:32	-	52%	99%	97%
☑	10.239.219.4 /var/lib/HPCCSystems/mythor	Thor Slave [mythor]	1	Normal	Ready	04:32	-	52%	99%	98%
☑	10.239.219.3 /var/lib/HPCCSystems/mythor	Thor Master		Normal	Ready	04:32	-	51%	99%	97%

☑ Select All / None

Fetched: 11/08/12 11:28:41

Action: Machine Information ▾

☑ Get processor information — Warn if CPU usage is over 95 %

☑ Get storage information — Warn if available memory is under 5 % ▾

☑ Local File Systems Only

☑ Get software information — Warn if available disk space is under 5 % ▾

☑ Show processes using filter

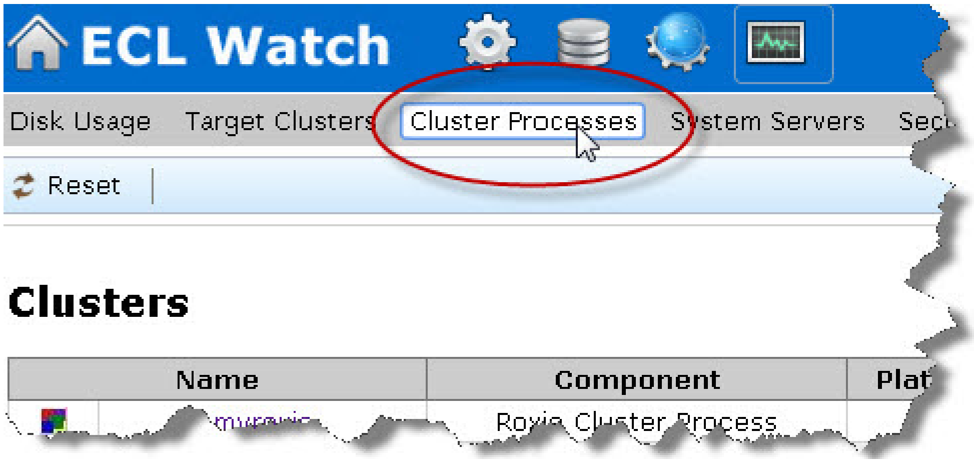
Additional processes to filter:

□ Auto Refresh every 0 mins.

Preflight the Roxie Cluster

1. Click on the **Operations** icon then click on the **Cluster Processes** link.

Figure 11. Cluster Processes Link

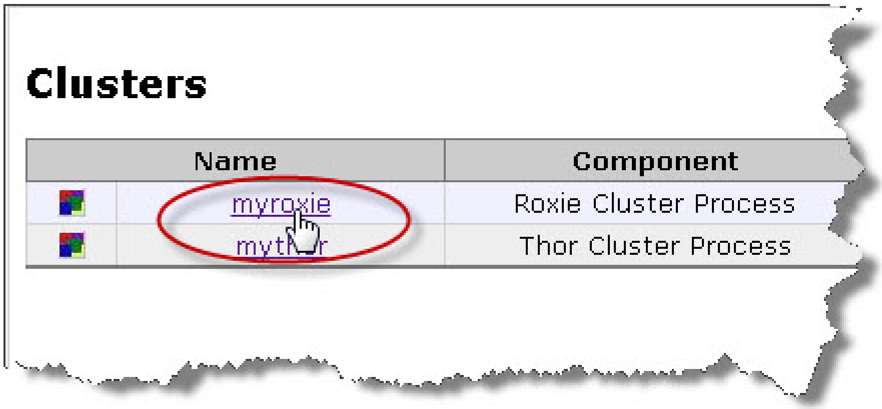


Clusters

	Name	Component	Plat
■	myroxie	Roxie Cluster Process	

2. Click on the **myroxie** link.

Figure 12. myroxie link



Clusters

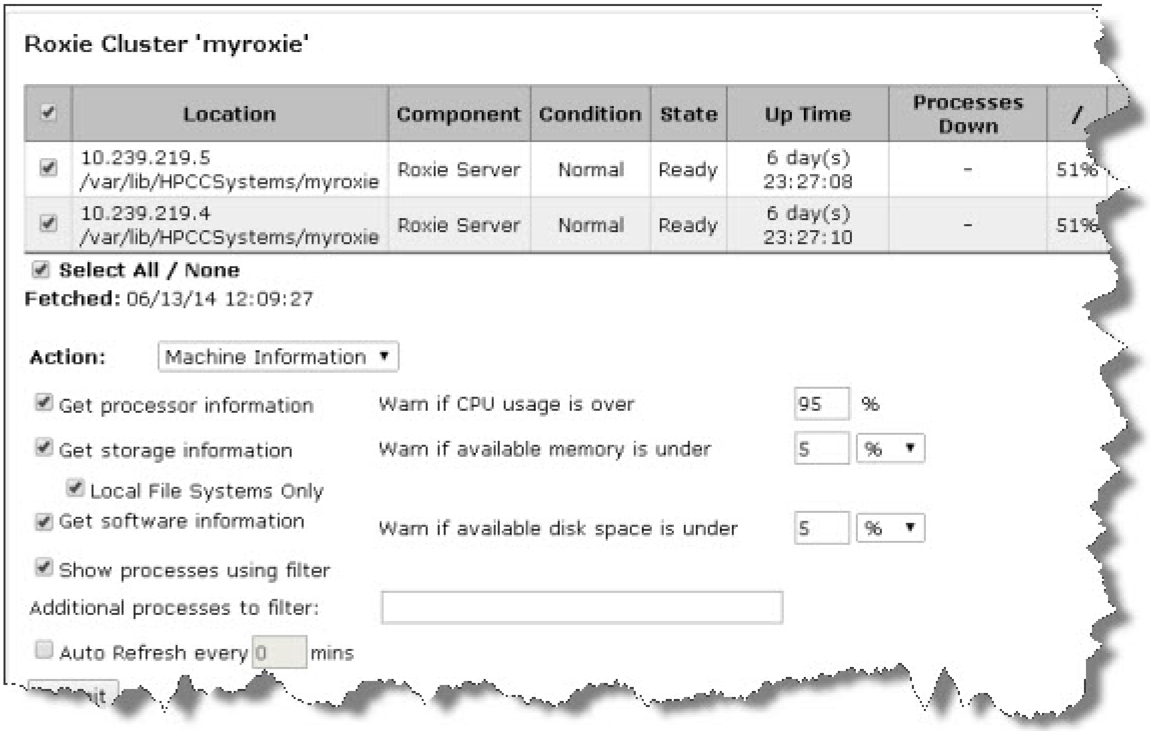
	Name	Component
■	myroxie	Roxie Cluster Process
■	mythor	Thor Cluster Process

3. Press the **Submit** button to start preflight.

EXPECTED RESULTS

After pressing Submit, a screen similar to the following should display.

Figure 13. Roxie system information



Roxie Cluster 'myroxie'

☑	Location	Component	Condition	State	Up Time	Processes Down	/
☑	10.239.219.5 /var/lib/HPCCSystems/myroxie	Roxie Server	Normal	Ready	6 day(s) 23:27:08	-	51%
☑	10.239.219.4 /var/lib/HPCCSystems/myroxie	Roxie Server	Normal	Ready	6 day(s) 23:27:10	-	51%

☑ Select All / None

Fetched: 06/13/14 12:09:27

Action: Machine Information ▾

☑ Get processor information — Warn if CPU usage is over: 95 %

☑ Get storage information — Warn if available memory is under: 5 % ▾

☑ Local File Systems Only

☑ Get software information — Warn if available disk space is under: 5 % ▾

☑ Show processes using filter

Additional processes to filter:

□ Auto Refresh every 0 mins

This indicates whether the Roxie nodes are up and running, and some information about them.

	If your system has more than 1 Roxie cluster, repeat these steps for each cluster.

Certify Thor & Roxie

The following sections will help you to Certify that the Thor, hThor, and Roxie components of your system are all working correctly.

Build Data on Thor

1. Open **the ECL IDE**

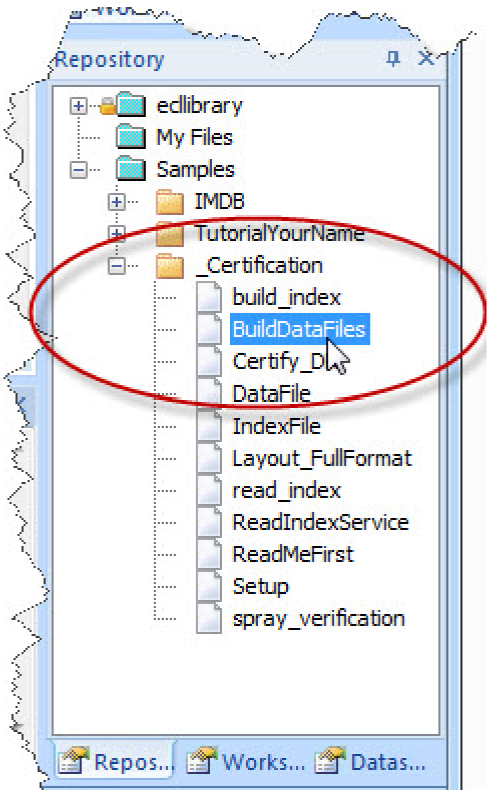
   Enter the Login ID and Password provided.

Login ID	hpccdemo
Password	hpccdemo

2. Open the **_Certification.BuildDataFiles** file.

   - In the lower right corner of the ECL IDE you will see a section labeled **Repository**, containing a few folders. These folders contain the ECL files. Click the + sign next to **Samples**, open the folder.
   - Navigate to the **_Certification** folder and click the + sign next to it to. Open it and view the contents.

   Figure 14. ECL Files in _Certification



   - Double-click on the **BuildDataFiles** file to open it.

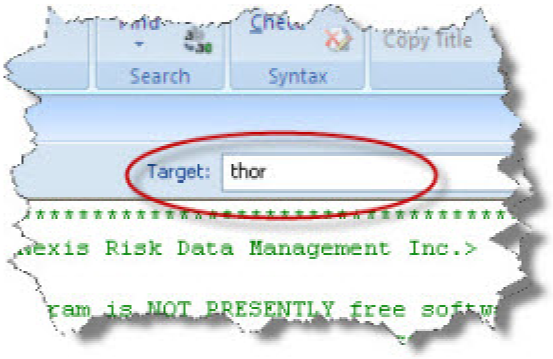
   In the BuildDataFiles file, you will see some ECL code in the file as follows:

```
Base := Cert.Setup.NodeMult1; //max = 20
Mult := Cert.Setup.NodeMult2; //max = 20
```

These two multipliers, **NodeMult1** and **NodeMult2** define the total number of millions of records. The values as configured in the _Certification sample generate 2,000,000 records. Typically you would want to generate 1 million records per node, up to 400 nodes. The maximum data set size is 18,800,000,000 bytes (47 * 400 million). The code used in this example is designed to generate a maximum of 400 million records. A larger number of nodes will result in fewer records per node, however the code will still work as intended for this exercise.

3. Select **thor** as Target from the drop menu on the right side.

Figure 15. Target thor

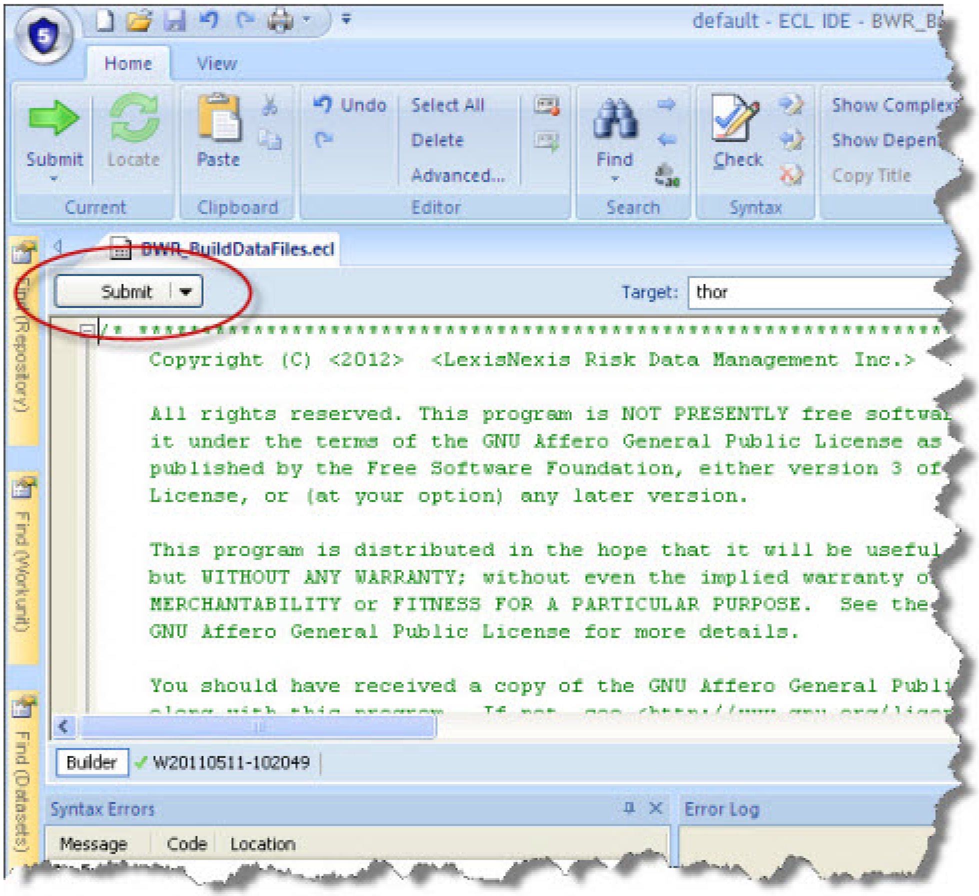


Target: thor

```
*************************************
exis Risk Data Management Inc.>

ram is NOT PRESENTLY free softw
```

4. Press the **Submit** button.

Figure 16. Submit button



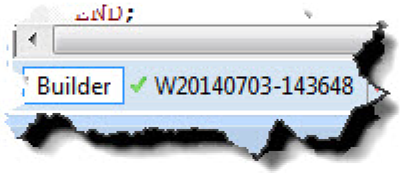
```
/* ***********************************************************************
   Copyright (C) <2012>  <LexisNexis Risk Data Management Inc.>

   All rights reserved. This program is NOT PRESENTLY free softwa
   it under the terms of the GNU Affero General Public License as
   published by the Free Software Foundation, either version 3 of
   License, or (at your option) any later version.

   This program is distributed in the hope that it will be useful
   but WITHOUT ANY WARRANTY; without even the implied warranty o
   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
   GNU Affero General Public License for more details.

   You should have received a copy of the GNU Affero General Publi
```

Note: This code generates a data file and writes it to disk.

EXPECTED RESULT:

Look for the green checkmark indicating successful completion

Figure 17. Green Checkmark

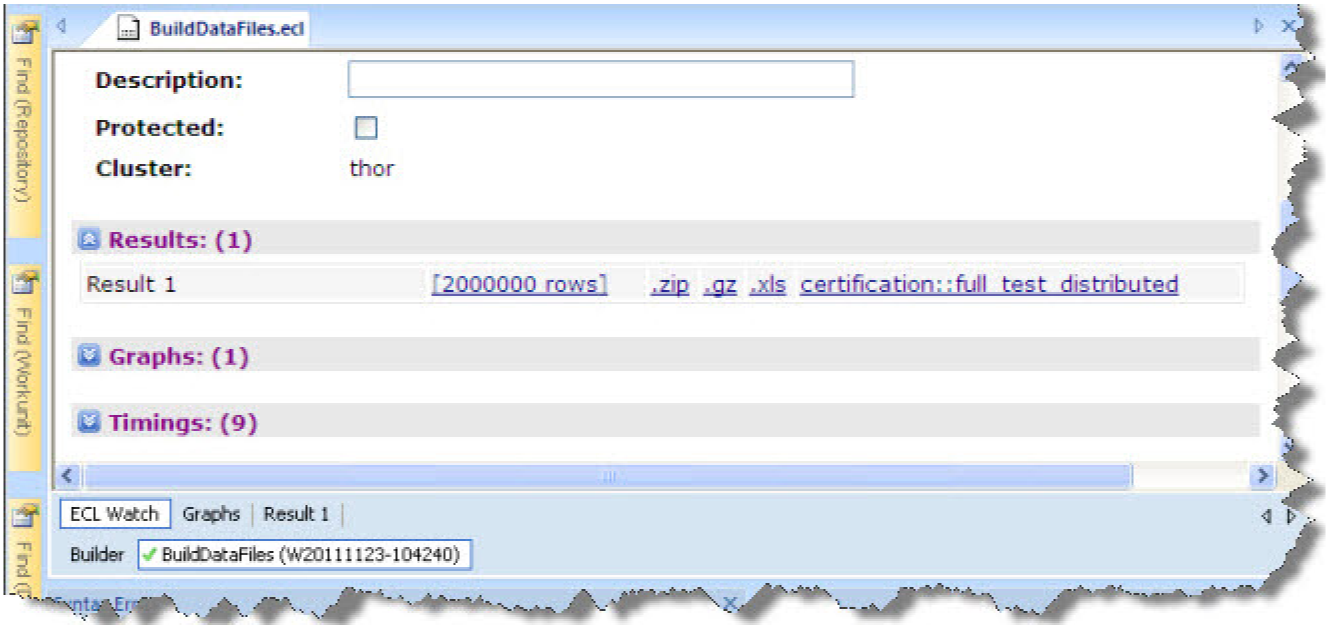


1. **Open ECLWatch** and look at the Workunit details page. This illustrates the steps of the query in real-time.

   This process generates 2,000,000 - 47-byte records in a file with the logical name of **certification::full_test_distributed**

   Note: The filename and other variables are defined in the **_Certification.Setup** file.

Figure 18. Workunit details page



Description:

Protected: □

Cluster: thor

Results: (1)

Result 1 [2000000 rows] .zip .gz .xls certification::full_test_distributed

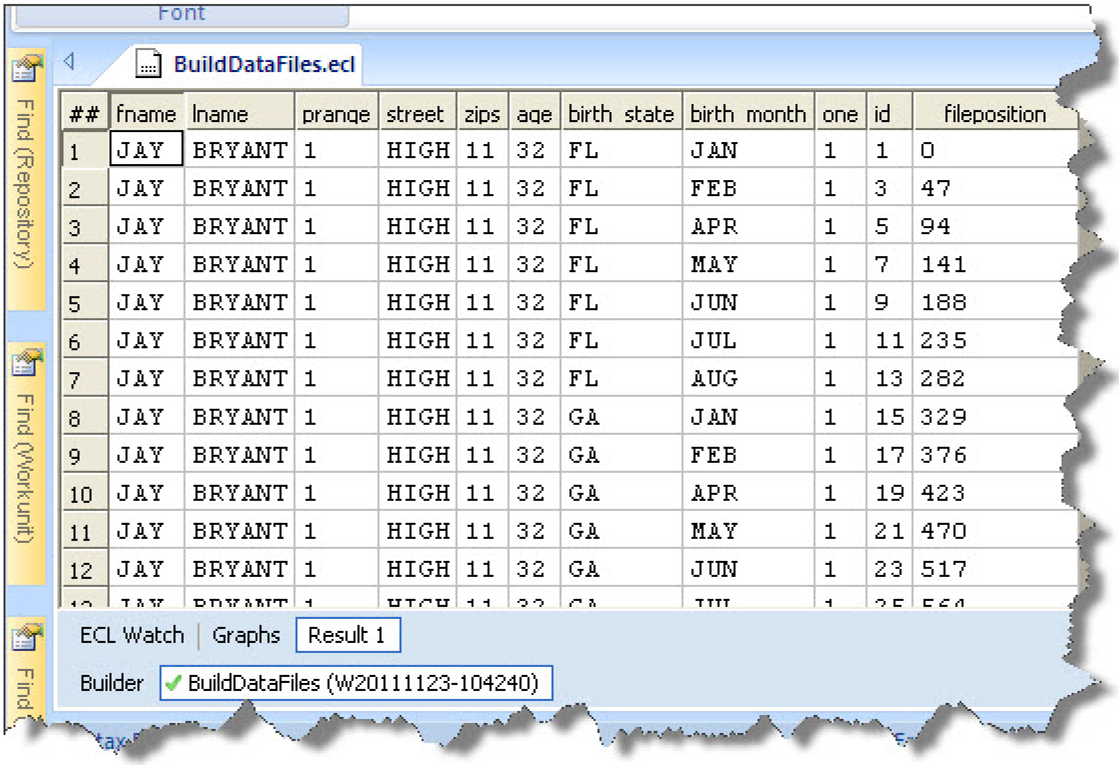
Graphs: (1)

Timings: (9)

Builder ✓ BuildDataFiles (W20111123-104240)

2. View the result by selecting the **Result1** tab (should be similar to the following):

Figure 19. View the Result



##	fname	lname	prange	street	zips	age	birth_state	birth_month	one	id	fileposition
1	JAY	BRYANT	1	HIGH	11	32	FL	JAN	1	1	0
2	JAY	BRYANT	1	HIGH	11	32	FL	FEB	1	3	47
3	JAY	BRYANT	1	HIGH	11	32	FL	APR	1	5	94
4	JAY	BRYANT	1	HIGH	11	32	FL	MAY	1	7	141
5	JAY	BRYANT	1	HIGH	11	32	FL	JUN	1	9	188
6	JAY	BRYANT	1	HIGH	11	32	FL	JUL	1	11	235
7	JAY	BRYANT	1	HIGH	11	32	FL	AUG	1	13	282
8	JAY	BRYANT	1	HIGH	11	32	GA	JAN	1	15	329
9	JAY	BRYANT	1	HIGH	11	32	GA	FEB	1	17	376
10	JAY	BRYANT	1	HIGH	11	32	GA	APR	1	19	423
11	JAY	BRYANT	1	HIGH	11	32	GA	MAY	1	21	470
12	JAY	BRYANT	1	HIGH	11	32	GA	JUN	1	23	517

Certify Thor Functionality

This section certifies:

- Certify Data Refinery full sort capabilities
- Certify Data Refinery local sort capabilities
- Certify Data Refinery local dedup capabilities
- Certify Data Refinery hash dedup capabilities
- Certify Data Refinery compress I/O capabilities
- Certify Data Refinery string search capabilities

Certify Thor

1. Open the **ECL IDE**

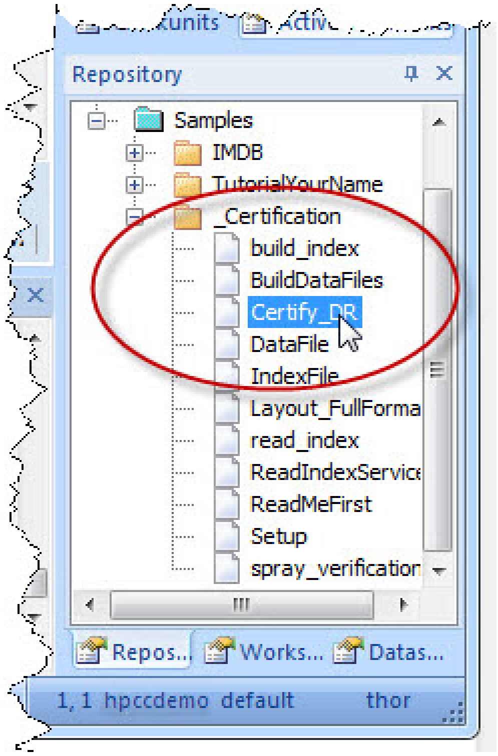
   Enter the Login ID and Password.

Login ID	hpccdemo
Password	hpccdemo

2. Open the **_Certification.Certify_DR** file.

   - In the lower right corner of the ECL IDE you will see a section labeled as Repository, containing a few folders. This contains the ECL files. Click the + sign next to *Samples*, to open the folder.
   - Navigate to the _Certification folder and click the + sign next to it to open it and view the contents.

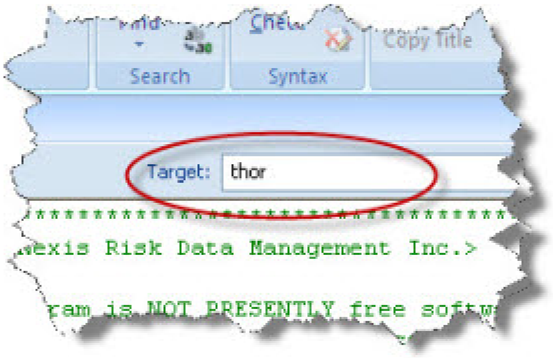
Figure 20. ECL Certify_DR File



- Double-click on the **Certify_DR** file to open it.

3. Select **thor** as the Target from the drop menu on the right side.

Figure 21. Target: thor

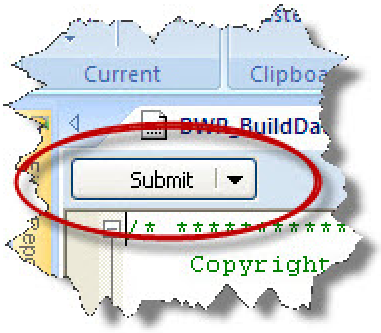


Target: thor

```
*********************************************
exis Risk Data Management Inc.>
ram is NOT PRESENTLY free softw
```

4. Press the **Submit** button.

Figure 22. Submit button

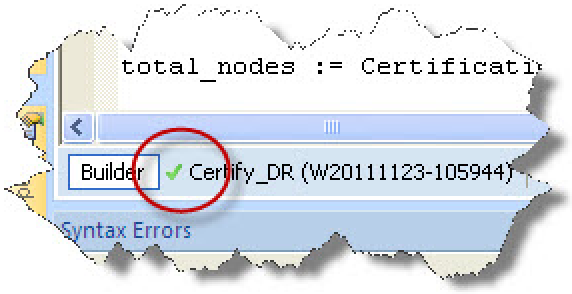


Note: This file uses the previously generated data and tests a series of Thor capabilities. It does not write data to disk. It will take a few minutes to complete, depending on the size of your system.

EXPECTED RESULT

Look for the Green checkmark indicating successful completion.

Figure 23. Green checkmark



The ECL Watch **Results** tab section should be as follows:

Result 1 Full Global Join - should = 2 million : 2000000

Result 2 Local Join - should = 2 million (local): 2000000

Result 3 Dedup - should = 2 million (joined): 2000000

Result 4 Complex I/O - should = 2 million: 2000000

Result 5 Hash Aggregate (Should be 2 records): 2

Result 6 Global Aggregate (Should be 2 records): 2

Result 7 Local Aggregate (Should be 2 records): 2

Result 8 Global Grouped Rollup (Should be 2 records): 2

Result 9 Local Rollup (Should be 2 records): 2

Result 10 Local Grouped Rollup (Should be 2 records): 2

Result 11 Global It/Srt/Ddp (Should be 2 records): 2

Result 12 Global Grouped It/Srt/Ddp (Should be 2 records): 2

Result 13 Local It/Srt/Ddp (Should be 2 records): 2

Result 14 Local Grouped It/Srt/Ddp (Should be 2 records): 2

Result 15 String Search Results: 100000

Certify Key build capabilities

This section will certify that the system can perform its key build capabilities.

1. Open the **ECL IDE**

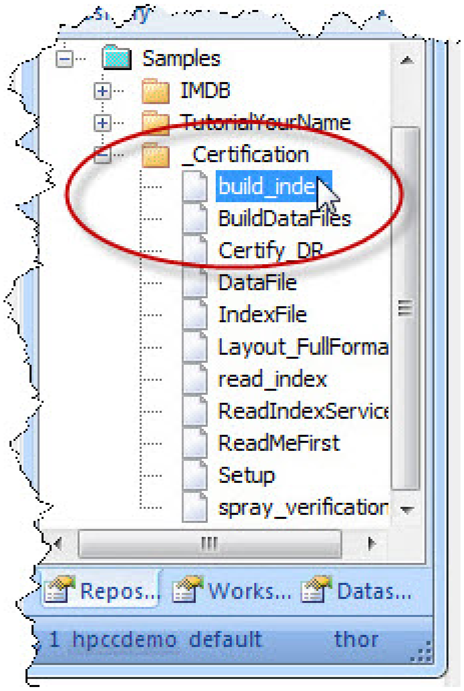
   Enter the Login ID and Password.

Login ID	hpccdemo
Password	hpccdemo

2. Open **_Certification.build_index** file.

   - In the lower right corner of the ECL IDE you will see a section labeled as Repository, containing a couple of folders. This contains the ECL files. Select the + sign next to it Samples, open the folder.
   - Navigate to the **_Certification** folder and select the + sign next to it to open it and view the contents.

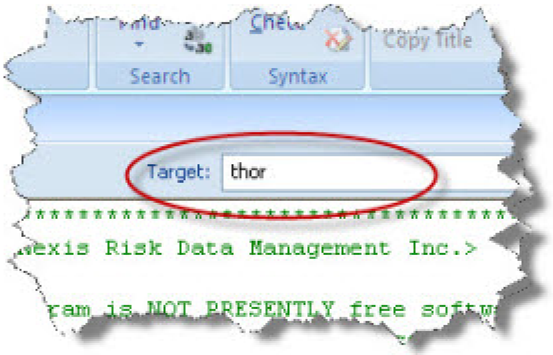
   Figure 24. Expand the _Certification folder



   - Double-click on the **build_index** file to open it.

3. Select **thor** as Target from the drop menu on the right side.

Figure 25. Target: thor

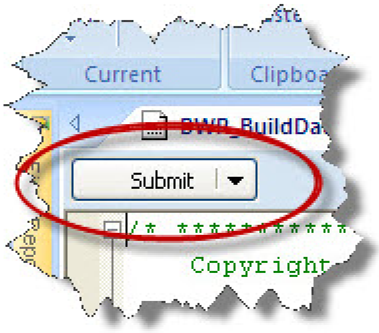


Target: thor

```
*****************************************
exis Risk Data Management Inc.>
ram is NOT PRESENTLY free softw
```

4. Press the **Submit** button.

Figure 26. Submit button



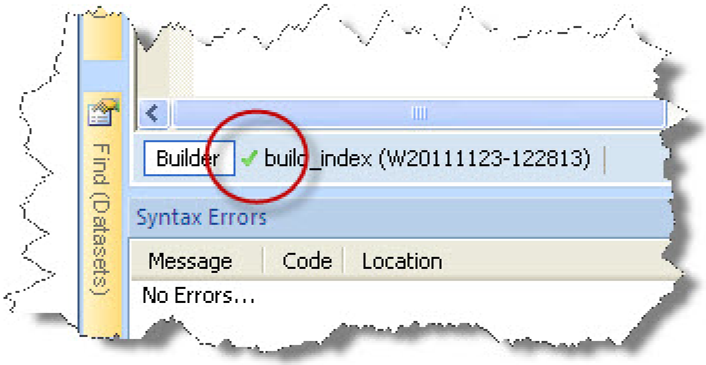
```
/* ***********
   Copyright
```

Note: This file uses the previously generated data. It builds an index on one of those data files.

EXPECTED RESULT

Look for the green checkmark indicating successful completion.

Figure 27. Green checkmark



✓ build_index (W20111123-122813)

Message	Code	Location
No Errors...		

The file we created earlier is indexed by Last Name and the index file, **thor::full_test_distributed_index**, is written to disk.

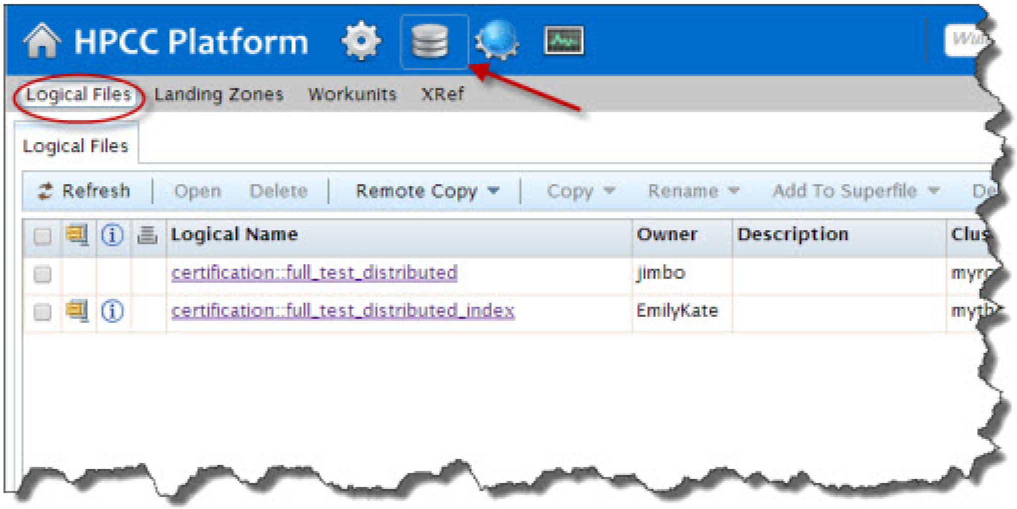
Verify the Index Build

1. Open ECL Watch in your browser using the following URL:

   http://nnn.nnn.nnn.nnn:pppp (where nnn.nnn.nnn.nnn is your ESP Server's IP Address and pppp is the port. The default port is 8010)

2. Click on the **Files** icon, then click on Logical Files.

   Figure 28. Browse Logical Files link

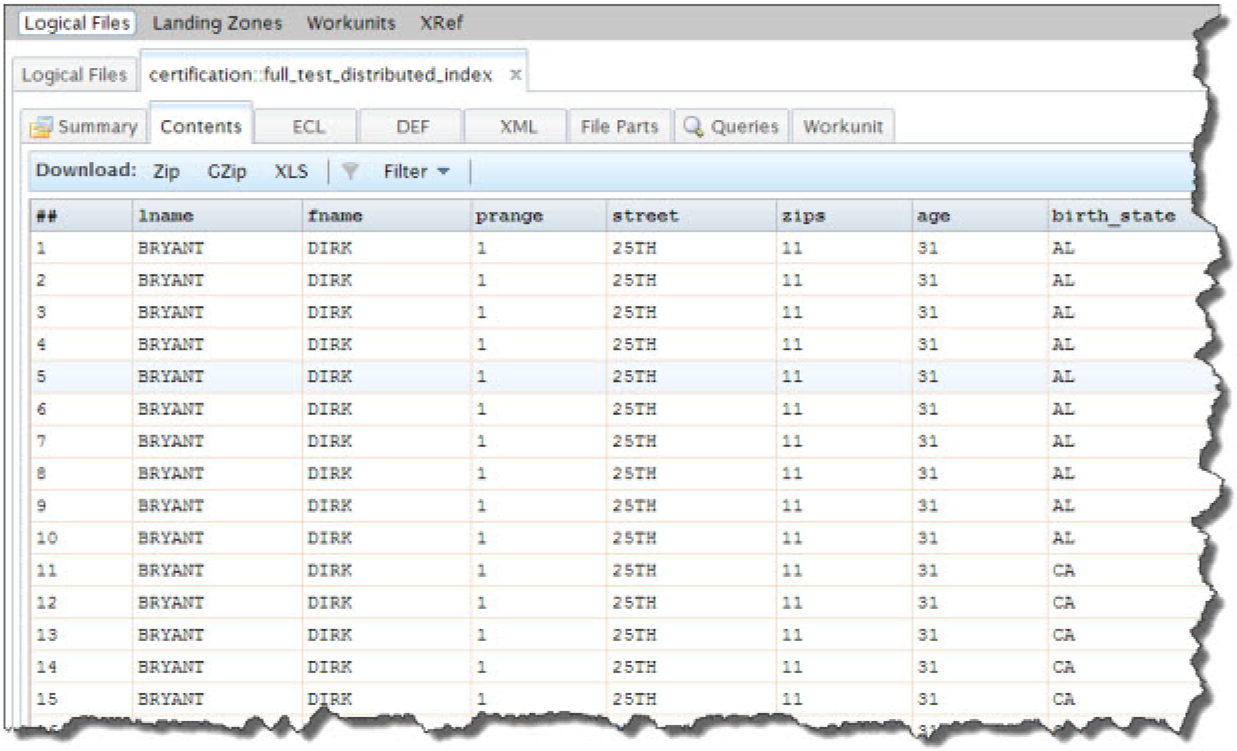


	Logical Name	Owner	Description	Clus
□	certification::full_test_distributed	jimbo		myr
□	certification::full_test_distributed_index	EmilyKate		myth

3. Check the box next to **certification::full_test_distributed_index** , then press the **Open** action button.

4. Select the Contents tab.

Figure 29. Contents



##	lname	fname	prange	street	zips	age	birth_state
1	BRYANT	DIRK	1	25TH	11	31	AL
2	BRYANT	DIRK	1	25TH	11	31	AL
3	BRYANT	DIRK	1	25TH	11	31	AL
4	BRYANT	DIRK	1	25TH	11	31	AL
5	BRYANT	DIRK	1	25TH	11	31	AL
6	BRYANT	DIRK	1	25TH	11	31	AL
7	BRYANT	DIRK	1	25TH	11	31	AL
8	BRYANT	DIRK	1	25TH	11	31	AL
9	BRYANT	DIRK	1	25TH	11	31	AL
10	BRYANT	DIRK	1	25TH	11	31	AL
11	BRYANT	DIRK	1	25TH	11	31	CA
12	BRYANT	DIRK	1	25TH	11	31	CA
13	BRYANT	DIRK	1	25TH	11	31	CA
14	BRYANT	DIRK	1	25TH	11	31	CA
15	BRYANT	DIRK	1	25TH	11	31	CA

Certify Thor Access to Indexed Data

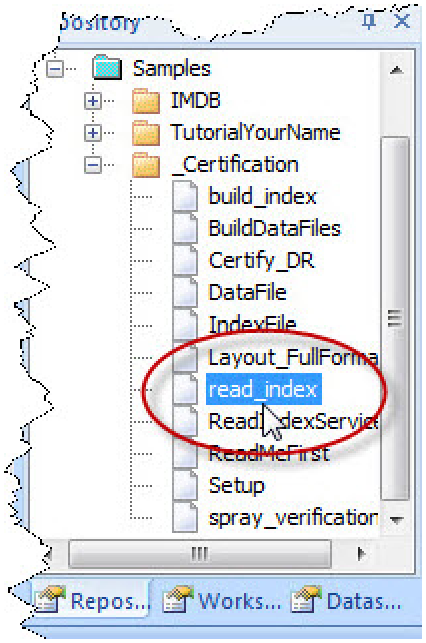
This section certifies Thor access to indexed data.

Certify Thor Access

1. Open the **_Certification.read_index** file.

   - In the lower right corner of the ECL IDE you will see a section labeled as Repository, containing a couple of folders. This contains the ECL files. Click the + sign next to *Samples*, open the folder.
   - Navigate to the **_Certification** folder and click the + sign next to it to open it and view the contents.

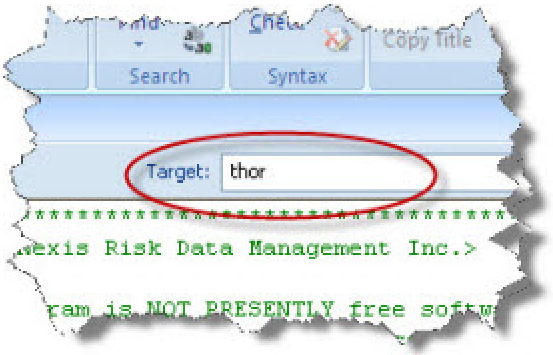
   Figure 30. Expand the _Certification folder

   - Double-click on the **read_index** file to open it.

2. Select **thor** as Target from the drop menu on the right side.

   Figure 31. Target: thor

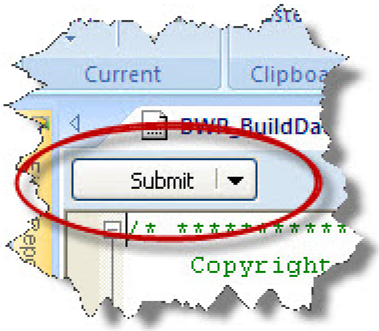
   

   Target: thor

   ```
   ****************************************************
   exis Risk Data Management Inc.>

   ram is NOT PRESENTLY free softw
   ```

3. Press the **Submit** button.

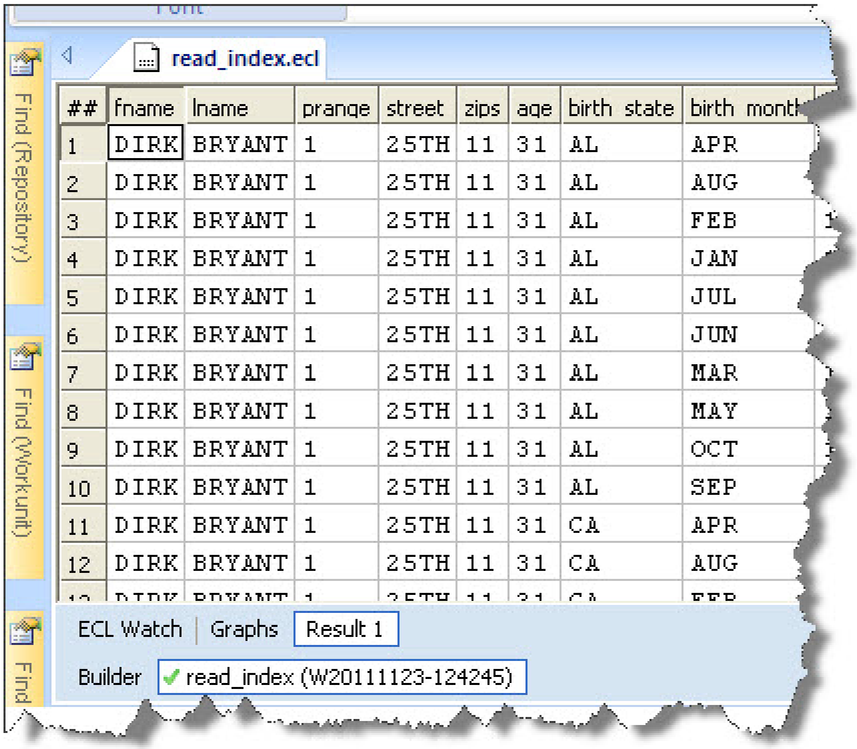
Figure 32. Submit button



EXPECTED RESULT:

The first 100 records from the query display, looking similar to the following (BRYANT in last name).

Figure 33. Results page



##	fname	lname	prange	street	zips	age	birth state	birth month
1	DIRK	BRYANT	1	25TH	11	31	AL	APR
2	DIRK	BRYANT	1	25TH	11	31	AL	AUG
3	DIRK	BRYANT	1	25TH	11	31	AL	FEB
4	DIRK	BRYANT	1	25TH	11	31	AL	JAN
5	DIRK	BRYANT	1	25TH	11	31	AL	JUL
6	DIRK	BRYANT	1	25TH	11	31	AL	JUN
7	DIRK	BRYANT	1	25TH	11	31	AL	MAR
8	DIRK	BRYANT	1	25TH	11	31	AL	MAY
9	DIRK	BRYANT	1	25TH	11	31	AL	OCT
10	DIRK	BRYANT	1	25TH	11	31	AL	SEP
11	DIRK	BRYANT	1	25TH	11	31	CA	APR
12	DIRK	BRYANT	1	25TH	11	31	CA	AUG
13	DIRK	BRYANT	1	25TH	11	31	CA	FEB

✓ read_index (W20111123-124245)

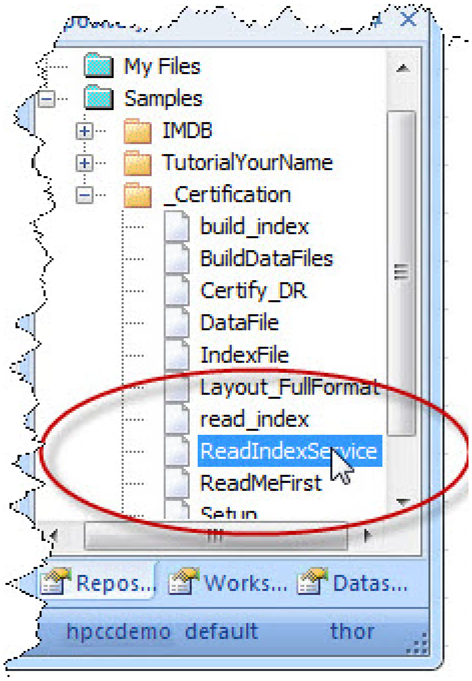
Compile and Publish a Roxie Query

1. Open the **_Certification.ReadIndexService** file, If you do not have it open already.

   - In the lower right corner of the ECL IDE you will see a section labeled as Repository, containing a couple of folders. This contains the ECL files. Click the + sign next to **Samples**, open the folder.

   - Navigate to the **_Certification** folder and click the + sign next to it to open it and view the contents.

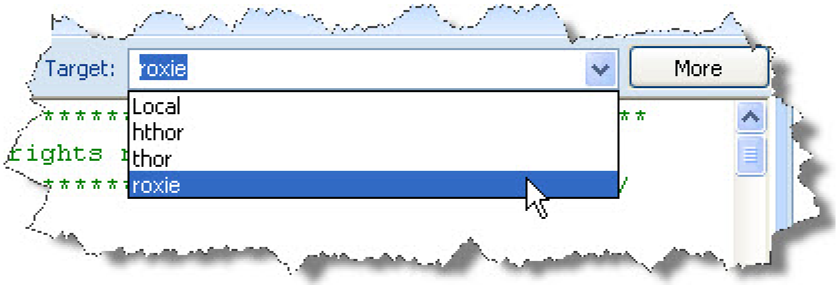
   Figure 34. _Certification folder

   - Double-click on the **ReadIndexService** file to open it.

2. Select **roxie** as Target from the drop menu on the right side.

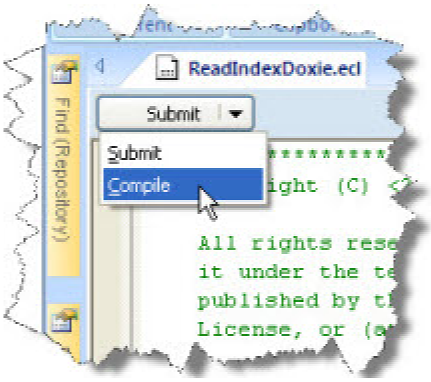
   Figure 35. Target roxie

3. In the upper left corner the **Submit** button has an arrow next to it. Select the arrow to expose the **Compile** option.

   Select **Compile** from the drop list under the submit button.

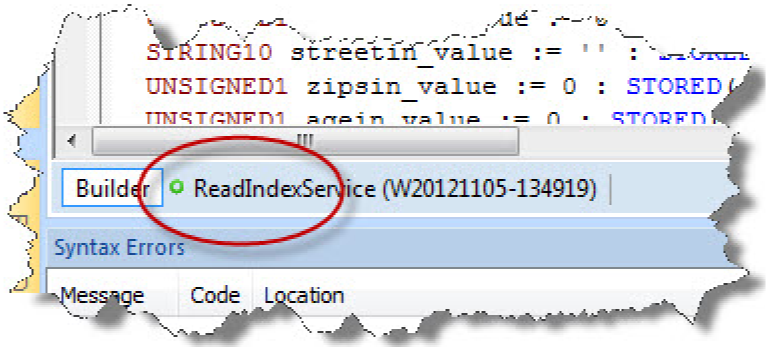
Figure 36. Compile



```
****************
ight (C)
All rights rese
it under the te
published by th
License, or (a
```

Look for a green circle indicating successful completion. Once complete, select the Workunit next to the the green circle.

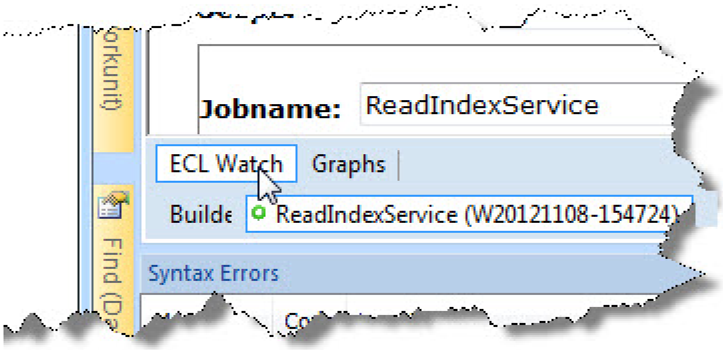
Figure 37. Completed Workunit: Green circle



```
STRING10 streetin_value := '' : STORED
UNSIGNED1 zipsin_value := 0 : STORED(
UNSIGNED1 agein_value := 0 : STORED(
```

4. Select the **ECL Watch** button at the lower left corner of the window.

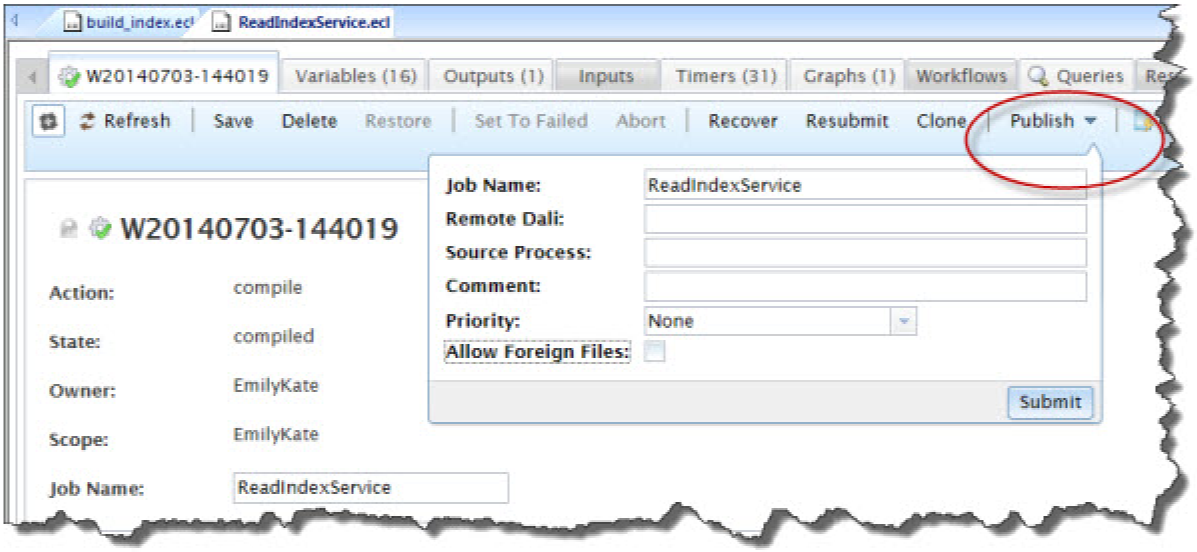
Figure 38. Select ECL Watch



Jobname: ReadIndexService

5. Select the **Publish** button from the ECL Watch tab that you just opened. (you may have to scroll down in the main window)

Figure 39. ECL Watch Publish button



W20140703-144019

Action: compile

State: compiled

Owner: EmilyKate

Scope: EmilyKate

Job Name: ReadIndexService

Job Name: ReadIndexService

Remote Dali:

Source Process:

Comment:

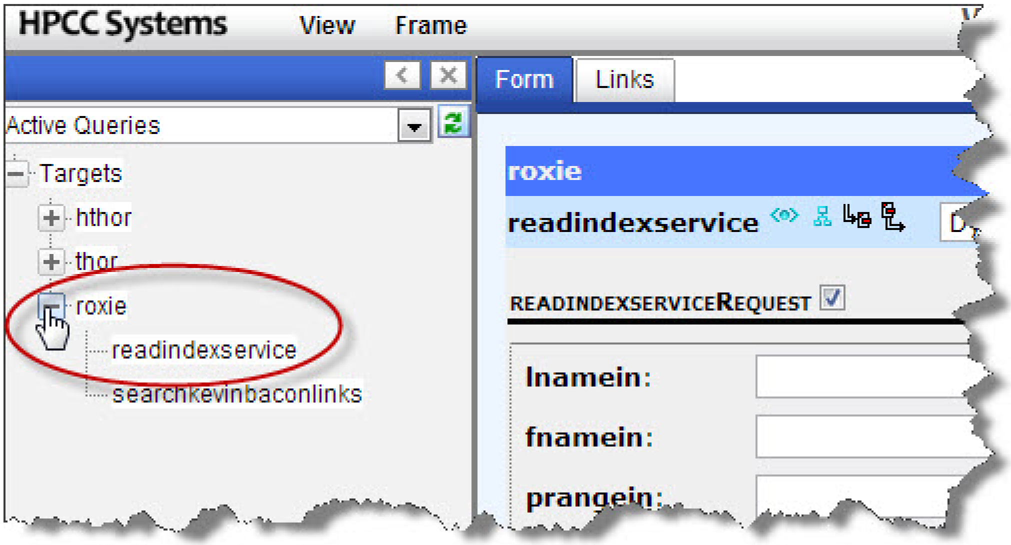
Priority: None

Allow Foreign Files: □

6. Open the ESP page in your browser using the following URL:

   http://nnn.nnn.nnn.nnn:pppp (where nnn.nnn.nnn.nnn is your ESP Server’s IP Address and pppp is the port. The default port is 8002)

Figure 40. Roxie ESP

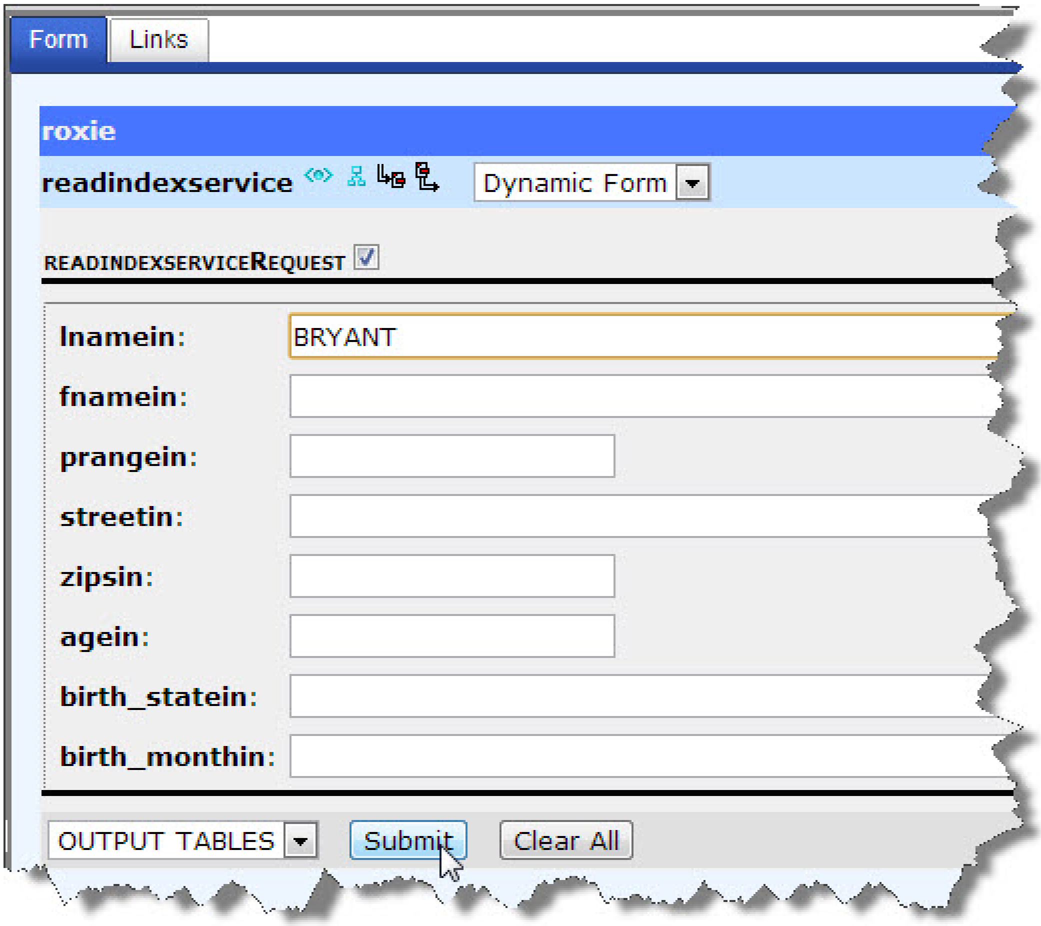


7. Click the + sign next to **roxie**, to expand it

8. Click **readindexservice**

9. Enter the name BRYANT in the **lname** field.

Figure 41. Enter lnamein



roxie

readindexservice

READINDEXSERVICEREQUEST ☑

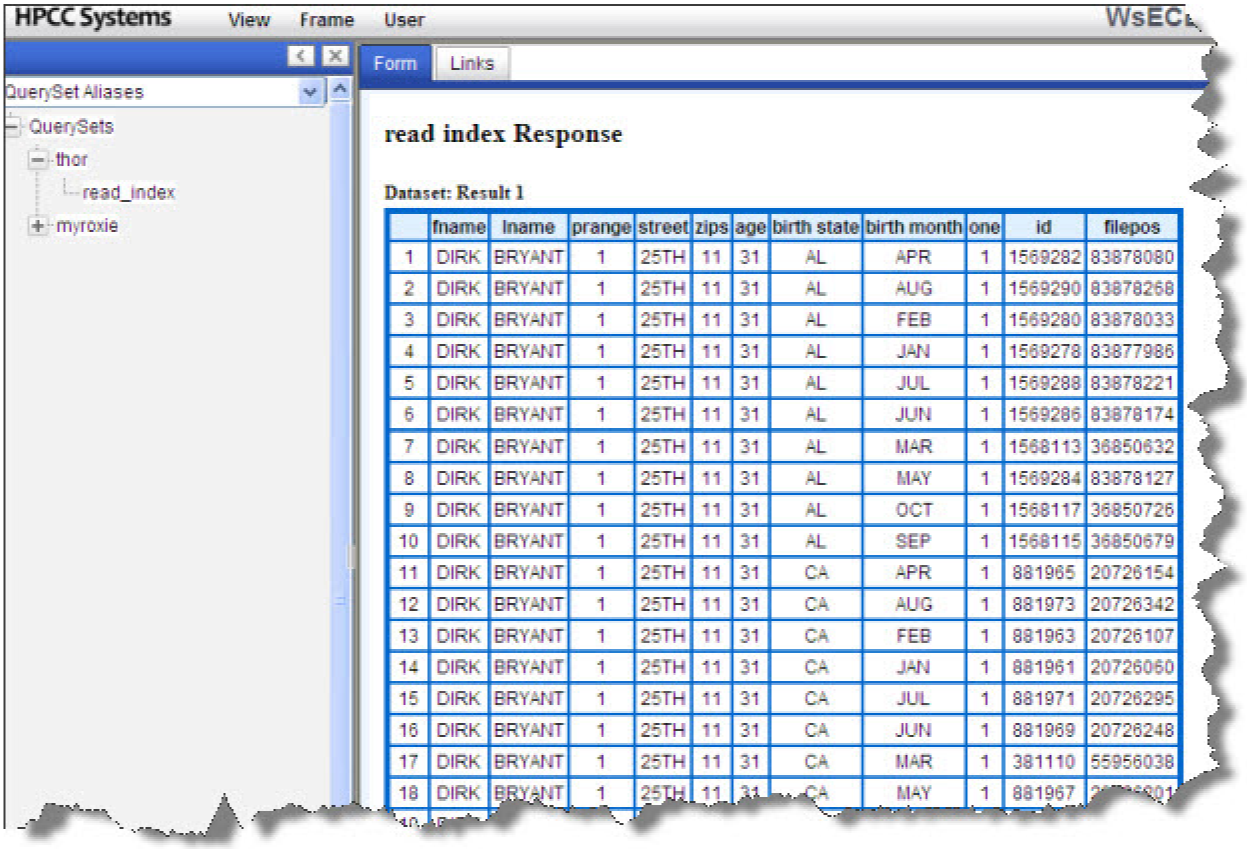
lnamein:	BRYANT
fnamein:	
prangein:	
streetin:	
zipsin:	
agein:	
birth_statein:	
birth_monthin:	

10.Press the **Submit** button at the bottom of the form.

EXPECTED RESULT:

A list of 100 records should display, looking similar to the following (BRYANT in last name).

Figure 42. Result



read index Response

Dataset: Result 1

	fname	lname	prange	street	zips	age	birth state	birth month	one	id	filepos
1	DIRK	BRYANT	1	25TH	11	31	AL	APR	1	1569282	83878080
2	DIRK	BRYANT	1	25TH	11	31	AL	AUG	1	1569290	83878268
3	DIRK	BRYANT	1	25TH	11	31	AL	FEB	1	1569280	83878033
4	DIRK	BRYANT	1	25TH	11	31	AL	JAN	1	1569278	83877986
5	DIRK	BRYANT	1	25TH	11	31	AL	JUL	1	1569288	83878221
6	DIRK	BRYANT	1	25TH	11	31	AL	JUN	1	1569286	83878174
7	DIRK	BRYANT	1	25TH	11	31	AL	MAR	1	1568113	36850632
8	DIRK	BRYANT	1	25TH	11	31	AL	MAY	1	1569284	83878127
9	DIRK	BRYANT	1	25TH	11	31	AL	OCT	1	1568117	36850726
10	DIRK	BRYANT	1	25TH	11	31	AL	SEP	1	1568115	36850679
11	DIRK	BRYANT	1	25TH	11	31	CA	APR	1	881965	20726154
12	DIRK	BRYANT	1	25TH	11	31	CA	AUG	1	881973	20726342
13	DIRK	BRYANT	1	25TH	11	31	CA	FEB	1	881963	20726107
14	DIRK	BRYANT	1	25TH	11	31	CA	JAN	1	881961	20726060
15	DIRK	BRYANT	1	25TH	11	31	CA	JUL	1	881971	20726295
16	DIRK	BRYANT	1	25TH	11	31	CA	JUN	1	881969	20726248
17	DIRK	BRYANT	1	25TH	11	31	CA	MAR	1	381110	55956038
18	DIRK	BRYANT	1	25TH	11	31	CA	MAY	1	881967	[illegible]

Spray and Despray Data

This section verifies the systems ability to Spray and Despray data.

Spraying takes a file and distributes pieces of it across the nodes. Despray is the opposite--the system combines the data from the multiple nodes into a single file.

Certify Despray

The next step to verify that your system is working properly is to test the Despray capabilities. Despray is when the system combines the data from the multiple clusters into a singular file that can be moved to the Landing Zone from the Data Refinery.

Despray from ECL Watch

Despray is the opposite of spraying, is a good way to certify that piece is working properly.

1. To despray, go to ECL Watch in a browser window.

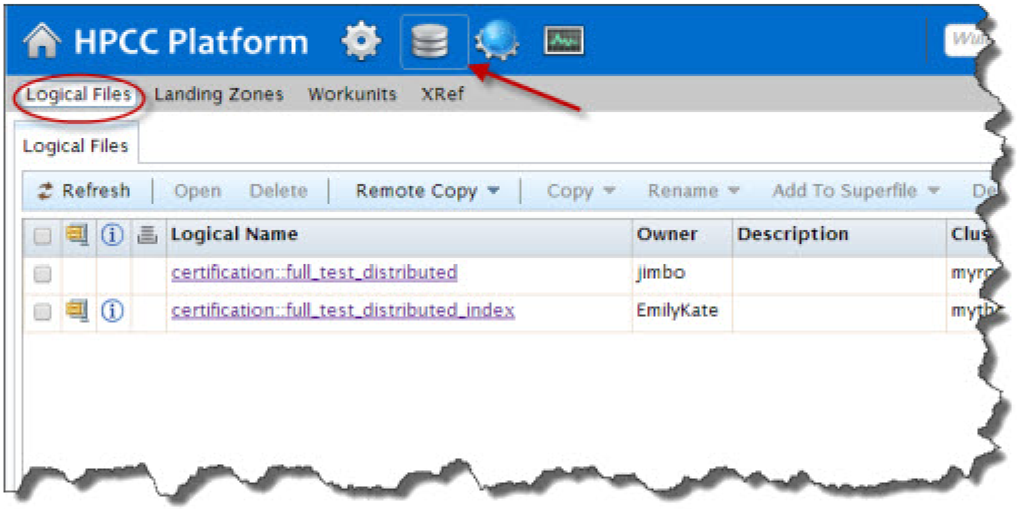
   Open ECL Watch in your browser using the following URL:

   http://nnn.nnn.nnn.nnn:pppp (where nnn.nnn.nnn.nnn is your ESP Server’s IP Address and pppp is the port. The default port is 8010)

	NOTE: To copy a URL to the clipboard, click the icon from the row of icons along the top of the ECL Watch page. This opens a window with the full URL. Select the entire URL and you can copy it to the clipboard to paste elsewhere.

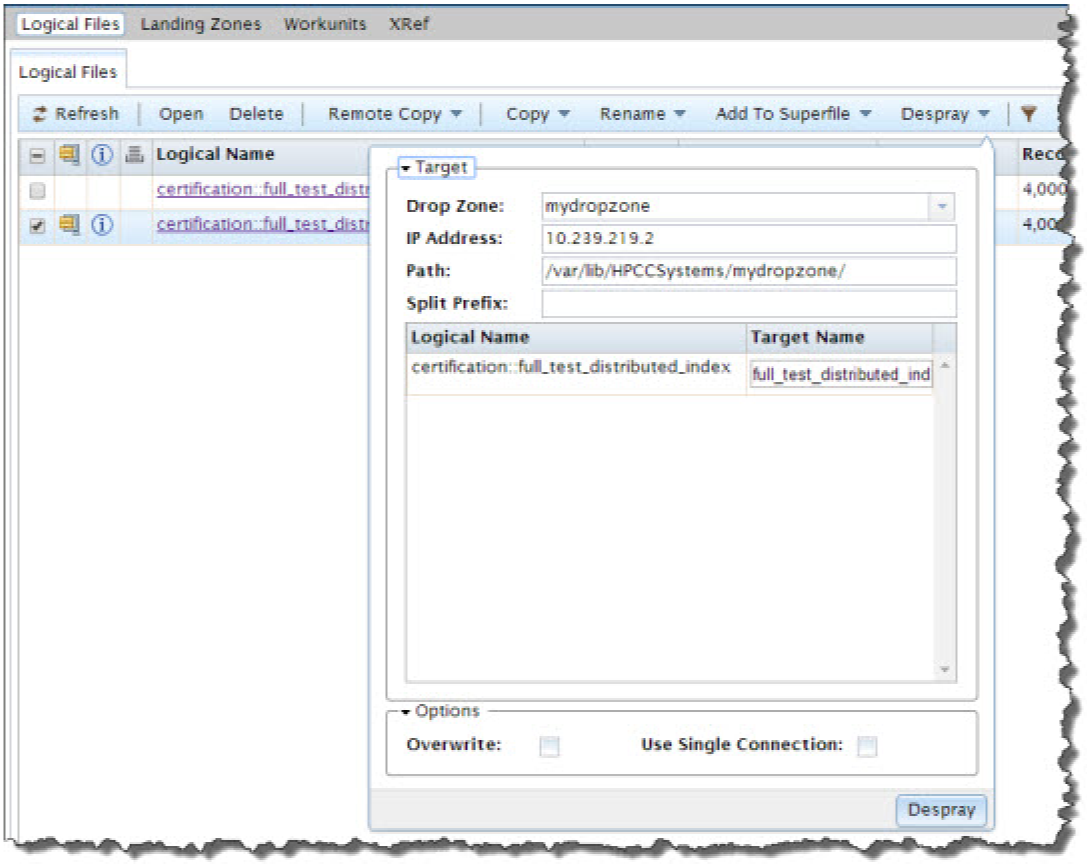
2. Click on the **Files** icon, then click on Logical Files.

   Figure 43. Browse Files



Logical Name	Owner	Description	Clus
certification::full_test_distributed	jimbo		myr
certification::full_test_distributed_index	EmilyKate		myth

3. Check the box next to **certification::full_test_distributed_index** , then press the **Despray** action button.



Drop Zone: mydropzone

IP Address: 10.239.219.2

Path: /var/lib/HPCCSystems/mydropzone/

Split Prefix:

Logical Name	Target Name
certification::full_test_distributed_index	full_test_distributed_ind

Overwrite: □ Use Single Connection: □

The **Despray File** dialog opens.

4. Provide **Destination** information.

Target

Drop Zone Use the drop list to select the machine to despray to. The items in the list are landing zones defined in the system's confguration. Your system may have only one.

IP Address This is prefilled based upon the selected machine.

Path The complete file path of the destination.

Split Prefix Prefix

Logical Name The Logical File to be sprayed (this is prefilled and cannot be altered)

Target Name The target filename. This is prefilled with the last portion of the Logical filename, but can be changed..

Overwrite Check this box to overwrite a file with the same name if it exists.

Use Single Connection Check this box to use a single network connection to despray the file.

5. Press the **Despray** button.

A DFU Workunit tab for each job opens. You can see the progress of each despray operation on the tab. If a job fails, information related to the cause of the failure also displays.

EXPECTED RESULTS:

Upon completion of the despray operation you will have a single file. You can then retrieve the file from the landing zone. This will certify that the despray operation is working correctly.

Certify Spray

The file will be sprayed from the Landing Zone to the Data Refinery, this will certify that data can be moved from Landing Zone to the Data Refinery.

Spray a Data File to your Thor Cluster

To use a data file in our HPCC cluster, we must first “spray” it to a Thor cluster. A *spray* or *import* is the relocation of a data file from one location to a Thor cluster. The term spray was adopted due to the nature of the file movement – the file is partitioned across all nodes within a cluster.

For this example, we will spray the full_test_distributed file that we just put out on our landing zone.

We are going to spray the file to our Thor cluster and give it a logical name of **certification::full_test_distributed**. The Distrubuted File Utility maintains a list of logical files and their corresponding physical file locations.

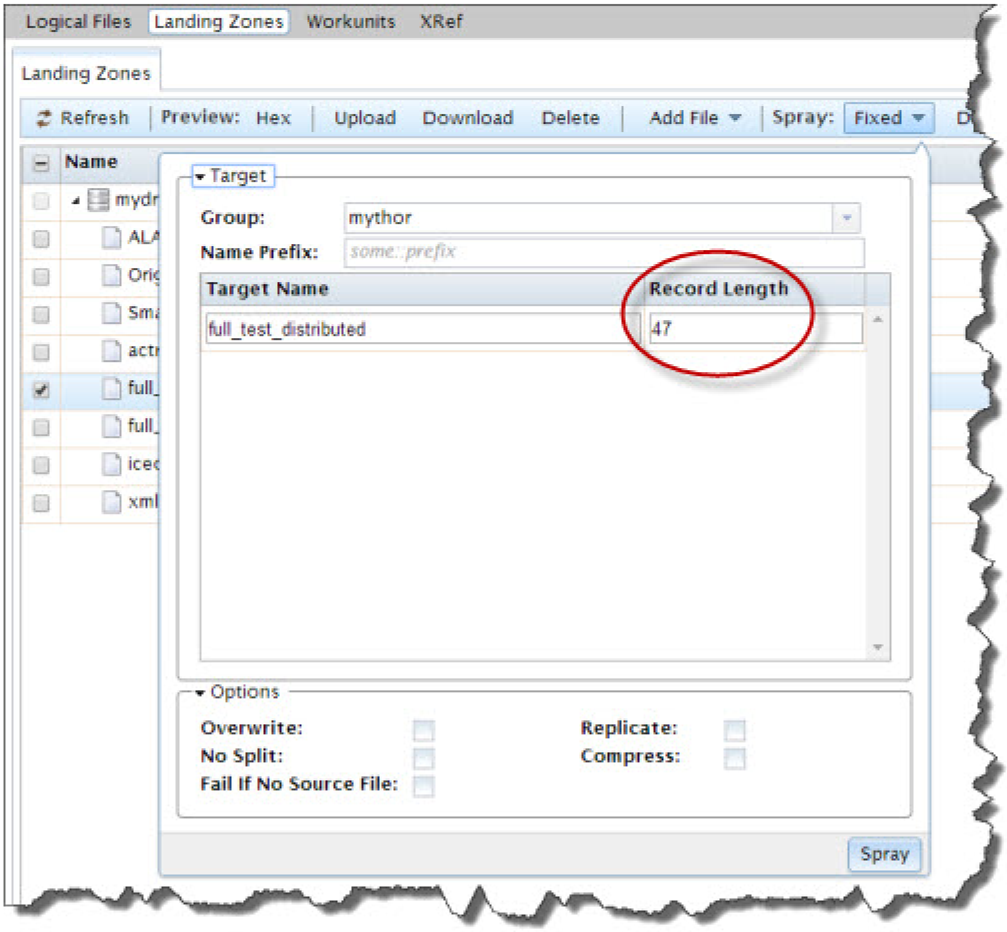
1. Click on the **Files** icon, then click the Landing Zones button on the navigation bar.

2. Click on the arrow next to your dropzone to expand the list.

   The files on your drop zone display.

3. Check the checkboxes for the file(s) you want to spray (*full_test_distributed*) , then click on the **Fixed** link.

   The **Spray Fixed** dialog displays.



Group: mythor

Name Prefix: some::prefix

Target Name	Record Length
full_test_distributed	47

Overwrite: □ Replicate: □

No Split: □ Compress: □

Fail If No Source File: □

4. Fill in relevant details:

Target

Group Select the name of cluster to spray to. You can only select a cluster in your environment.

Name Prefix The prefix for the logical file, in this case **certification**

Target Name The logical filename to create. This is pre-filled with the name of the source file on the landing zone, but can be changed.

Record Length The size of each record. In this case it is **47**

Options:

Overwrite Check this box to overwrite files of the same name.

Replicate Check this box to create backup copies of all file parts in the backup directory (by convention on the secondary drive of the node following in the cluster).

This option is only available on systems where replication has been enabled.

Compress Check this box to compress the files.

No Split Check this box to prevent splitting file parts to multiple target parts.

Fail if no source file Check this box to allow the spray to fail if no source file is found.

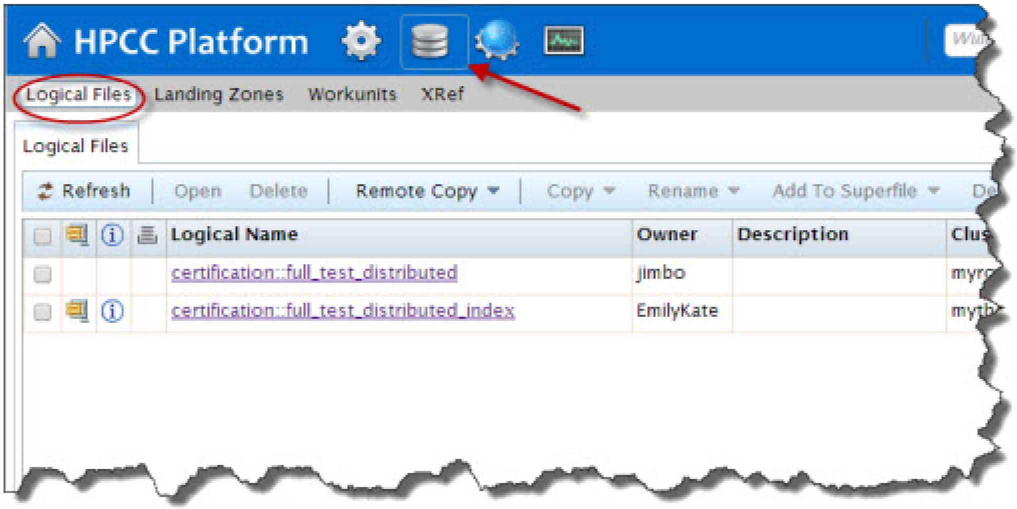
5. Press the **Spray** button.

   A **DFU Workunit** tab displays for each job. You can see the progress of each despray operation on the tab. If a job fails, information related to the cause of the failure also displays.

EXPECTED RESULTS

1. Click on the **Files** icon, then click on Logical Files.

Figure 44. Browse Files



				Logical Name	Owner	Description	Clus
□				certification::full_test_distributed	jimbo		myr
□		ⓘ		certification::full_test_distributed_index	EmilyKate		myth

2. Click on the sprayed file, select **Open** to view the logical file details..

3. Select the Contents tab to view contents.