

# **HPCC Systems® Preflight and Certification**

**Boca Raton Documentation Team**



---

# HPCC Systems® Preflight and Certification

Boca Raton Documentation Team

Copyright © 2026 HPCC Systems®. All rights reserved

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.

LexisNexis and the Knowledge Burst logo are registered trademarks of Reed Elsevier Properties Inc., used under license.

HPCC Systems® is a registered trademark of LexisNexis Risk Data Management Inc.

Other products, logos, and services may be trademarks or registered trademarks of their respective companies.

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

2026 Version 10.6.22-1

---

|                                             |    |
|---------------------------------------------|----|
| Introduction : Certification Approach ..... | 4  |
| Scope .....                                 | 5  |
| Before You Begin .....                      | 6  |
| Preflight .....                             | 7  |
| Preflight System Servers .....              | 8  |
| Preflight Target Clusters .....             | 12 |
| Preflight Thor .....                        | 16 |
| Preflight the Roxie Cluster .....           | 19 |
| Certify Thor & Roxie .....                  | 22 |
| Build Data on Thor .....                    | 22 |
| Certify Thor Functionality .....            | 27 |
| Certify Key build capabilities .....        | 31 |
| Verify the Index Build .....                | 33 |
| Certify Thor Access to Indexed Data .....   | 35 |
| Compile and Publish a Roxie Query .....     | 37 |
| Spray and Despray Data .....                | 42 |
| Certify Despray .....                       | 43 |
| Certify Spray .....                         | 46 |

# Introduction : Certification Approach

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

Use this procedure when you create a new HPCC Systems instance 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)**

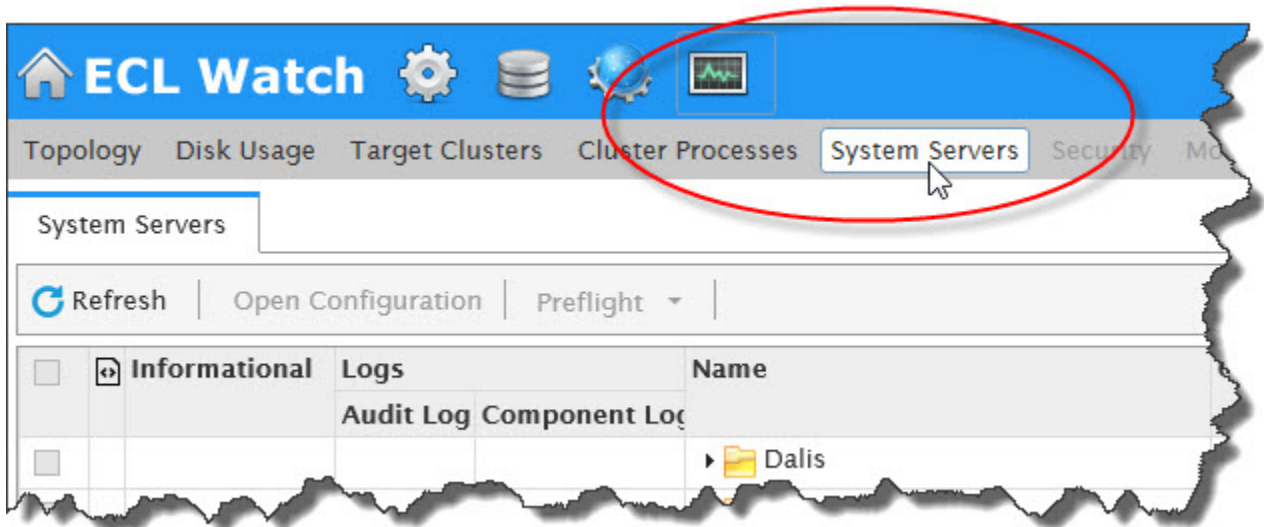


**Note:** That your IP address could be different from the ones provided in these figures. Please use the IP address provided by your installation.

# Preflight System Servers

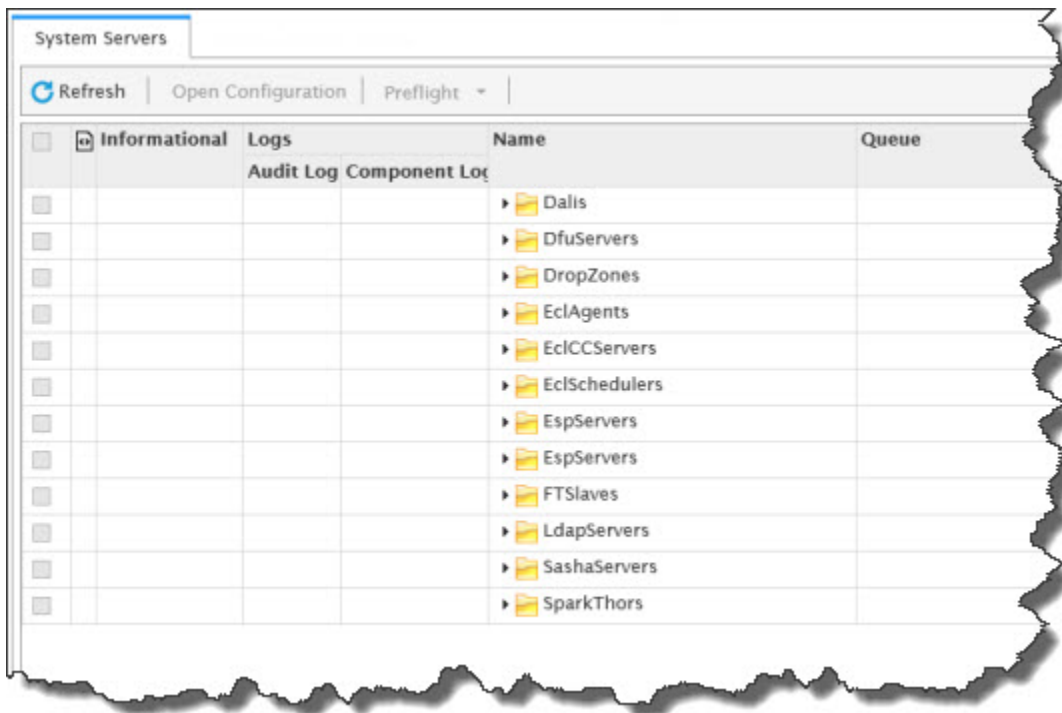
1. Click on the **Operations** icon then click on the **System Servers** link.

**Figure 1. System Servers link**



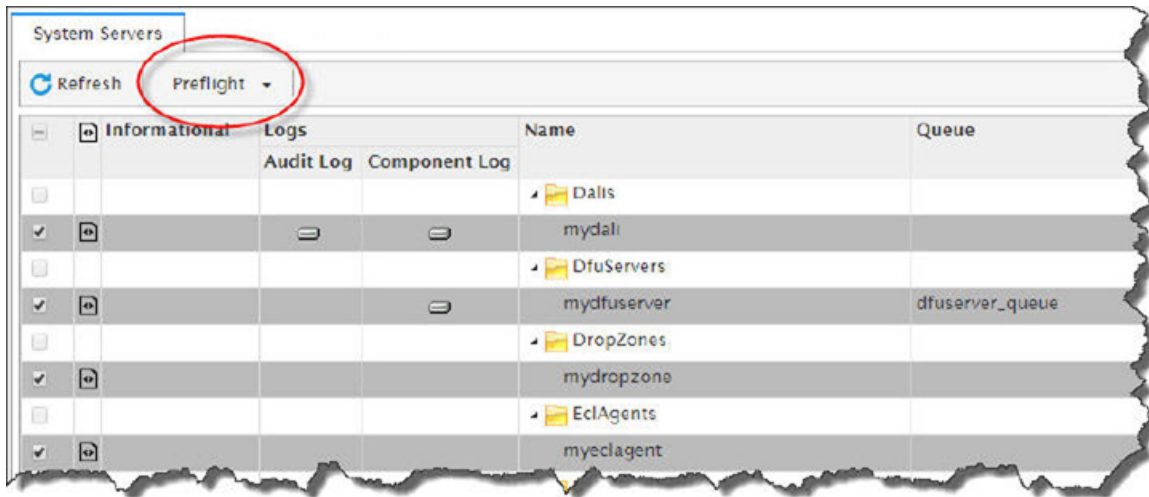
A screen similar to the following displays.

**Figure 2. System Servers page**



2. Expand the folder for the System Server then check the box next to the desired component(s).

**Figure 3. Select System Servers**



With the servers selected, the preflight action button activates and you can press it to display the preflight options.

3. Check or uncheck any desired options then Press the **Submit** button to start preflight.

**Figure 4. Submit**

rs (leg...

ion | Preflight ▾

Action: Machine Information ▾

Processor Information: ☒

Storage Information: ☒

Local File Systems Only: ☒

Get Software Information: ☒

Show Processes Using Filter: ☒

Additional Processes To Filter: Any Additional Processes To

Auto Refresh: ☒

Auto Refresh Increment: 5

Warn if CPU usage is over: 95

Warn if available memory is under: 95

Warn if available disk space is under: 95

Submit

01 - EcAgentProcess - eclagent node160101 Linux

## EXPECTED RESULTS:

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

**Figure 5. System Component Information**

TopologyDisk UsageTarget ClustersCluster ProcessesSystem ServersSecurityMonitoringDynamic ESOLog Visualization

System ServersMachine Information

Preflight Results

Refresh

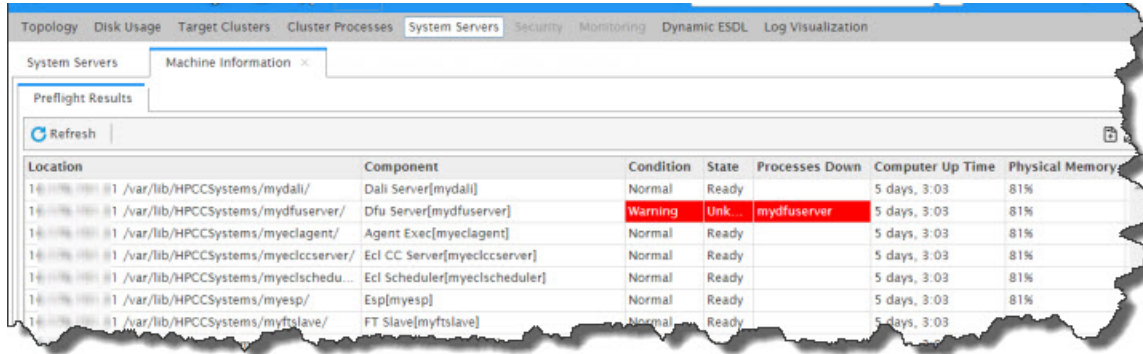
| Location                                        | Component               | Condition | State | Processes Down | Computer Up Time | Physical M |
|-------------------------------------------------|-------------------------|-----------|-------|----------------|------------------|------------|
| 10.176.151.31 /var/lib/HPCCSystems/mydali/      | Dali Server[mydali]     | Normal    | Ready |                | 4 days, 23:06    | 76%        |
| 10.176.151.31 /var/lib/HPCCSystems/mydfuserver/ | Dfu Server[mydfuserver] | Normal    | Ready |                | 4 days, 23:06    | 76%        |
| 10.176.151.31 /var/lib/HPCCSystems/myeclagent/  | Agent Exec[myeclagent]  | Normal    | Ready |                | 4 days, 23:06    | 76%        |
| 10.176.151.31 /var/lib/HPCCSystems/myeclagent/  | Agent Exec[myeclagent]  | Normal    | Ready |                | 4 days, 23:06    | 76%        |

This screen displays information about the selected system components. This information indicates whether the components are actually running appropriately. The resulting page shows useful information about each component. The component name, location, 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 alerts, the component(s) are highlighted, indicating they require further attention.

For example, the following image indicates there is an issue with the DFU Server.

**Figure 6. System Server Alert**



The screenshot shows the 'System Servers' tab in the Preflight interface. A table lists various system components. The row for 'Dfu Server[mydfuserver]' is highlighted in red, indicating a warning. The 'Condition' column for this row shows 'Warning', and the 'State' column shows 'Unk...'. The 'Processes Down' column for this row shows 'mydfuserver'.

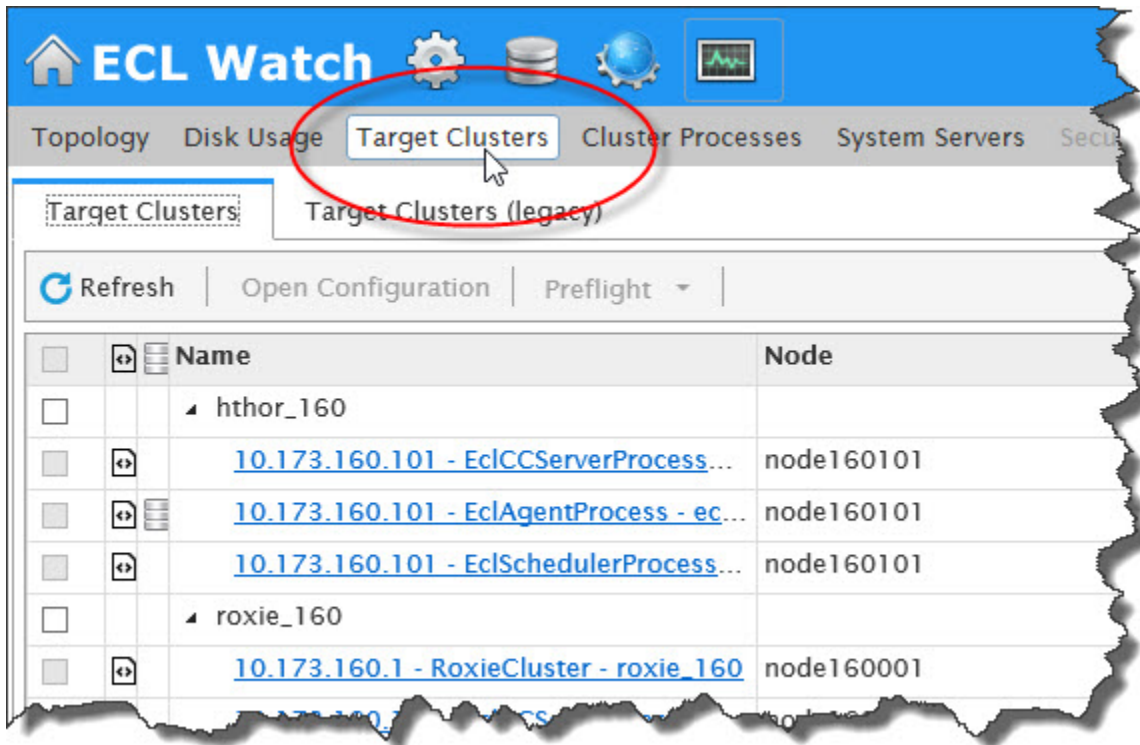
| Location                                          | Component                     | Condition | State  | Processes Down | Computer Up Time | Physical Memory |
|---------------------------------------------------|-------------------------------|-----------|--------|----------------|------------------|-----------------|
| 16.100.100.1 /var/lib/HPCCSystems/mydali/         | Dali Server[mydali]           | Normal    | Ready  |                | 5 days, 3:03     | 81%             |
| 16.100.100.1 /var/lib/HPCCSystems/mydfuserver/    | Dfu Server[mydfuserver]       | Warning   | Unk... | mydfuserver    | 5 days, 3:03     | 81%             |
| 16.100.100.1 /var/lib/HPCCSystems/myeclagent/     | Agent Exec[myeclagent]        | Normal    | Ready  |                | 5 days, 3:03     | 81%             |
| 16.100.100.1 /var/lib/HPCCSystems/myeclccserver/  | Ecl CC Server[myeclccserver]  | Normal    | Ready  |                | 5 days, 3:03     | 81%             |
| 16.100.100.1 /var/lib/HPCCSystems/myeclscheduler/ | Ecl Scheduler[myeclscheduler] | Normal    | Ready  |                | 5 days, 3:03     | 81%             |
| 16.100.100.1 /var/lib/HPCCSystems/myesp/          | Esp[myesp]                    | Normal    | Ready  |                | 5 days, 3:03     | 81%             |
| 16.100.100.1 /var/lib/HPCCSystems/myftslave/      | FT Slave[myftslave]           | Normal    | Ready  |                | 5 days, 3:03     |                 |

# Preflight Target Clusters

Use the Target Clusters link to preflight all your clusters.

1. Click on the **Operations** icon then click on the **Target Clusters** link.

**Figure 7. Target Clusters Link**



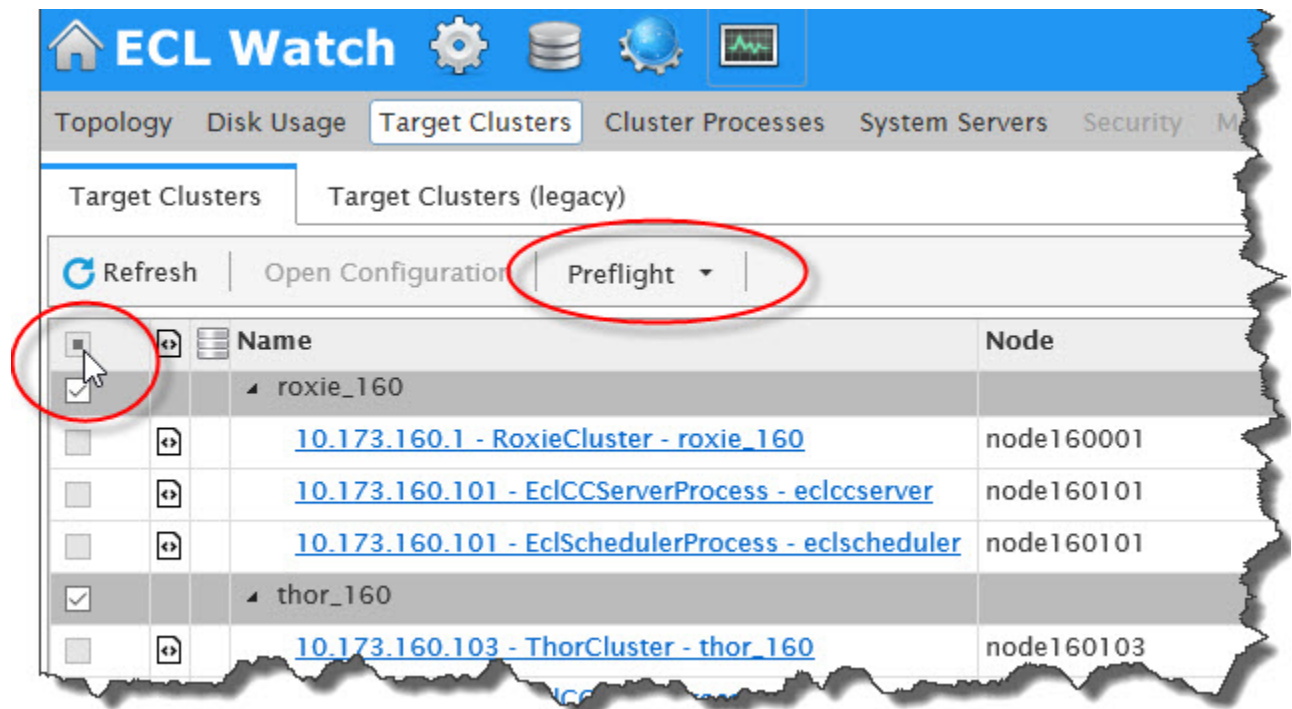
This displays a detailed listing of all your systems' Clusters.

2. Click on the select all check box, in the top row on the left side, to select all of the target clusters.

Optionally, you can just check the box(es) next to only the cluster(s) you want to preflight. If you choose to preflight all Target Clusters, you do not need to preflight Thor and Roxie separately as detailed below.

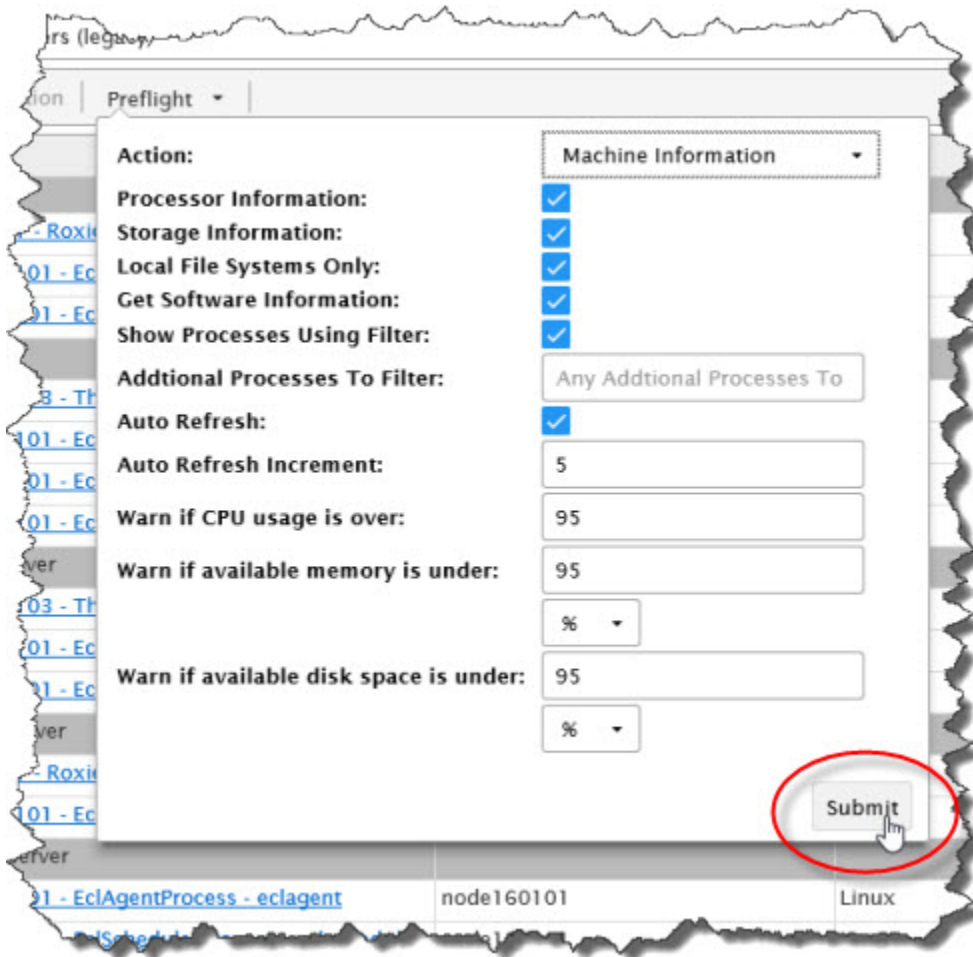
With the clusters selected, the preflight action button activates and you can press it to display the preflight options.

Figure 8. Select Target Clusters



3. Select or de-select any desired options, then press the **Submit** button at the bottom to start preflight.

Figure 9. Submit



**NOTE:** Depending on the size of your system, there could be a slight delay in displaying the results.

## EXPECTED RESULTS:

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

Figure 10. Target Cluster Information

The screenshot shows the 'Target Clusters' tab in the HPCC Systems interface. It displays a table of preflight results for various components across different nodes. The table includes columns for Location, Component, Computer Up Time, Physical Memory, Swap, and two additional metrics. The data is as follows:

| Location                              | Component                  | Computer Up Time | Physical Mem | Swap | /var/lib/HPCCSys | /run |
|---------------------------------------|----------------------------|------------------|--------------|------|------------------|------|
| 10.173.160.101 /var/lib/HPCCSystem... | Ecl CC Server[eclccserver] | 96 days, 17:03   | 80%          | 99%  | 80%              | 95%  |
| 10.173.160.101 /var/lib/HPCCSystem... | Agent Exec[eclagent]       | 96 days, 17:03   | 80%          | 99%  | 80%              | 95%  |
| 10.173.160.101 /var/lib/HPCCSystem... | Ecl Scheduler[eclsched...  | 96 days, 17:03   | 80%          | 99%  | 80%              | 95%  |
| 10.173.160.1 /var/lib/HPCCSystems...  | Roxie Server[roxie_160]    | 132 days, 23:44  | 20%          |      | 31%              | 91%  |
| 10.173.160.2 /var/lib/HPCCSystems...  | Roxie Server[roxie_160]    | 132 days, 23:44  | 20%          |      | 31%              | 91%  |
| 10.173.160.3 /var/lib/HPCCSystems...  | Roxie Server[roxie_160]    | 132 days, 23:44  | 20%          |      | 31%              | 91%  |

This screen displays information on your system's component nodes. This information can help to indicate if everything is operating normally or can help to point out any potential concerns.

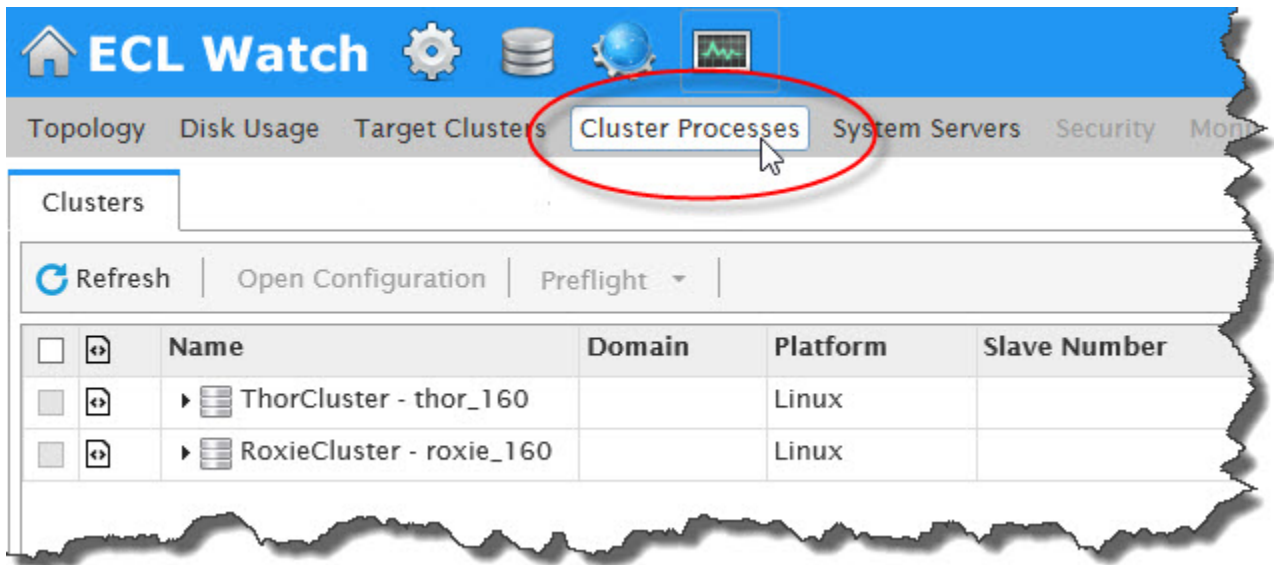
If there are any notable alerts, they are highlighted. These alerts usually require some attention.

If you have any alerts you should examine the specified component further. It is indicative of some kind of problem or abnormality.

# Preflight Thor

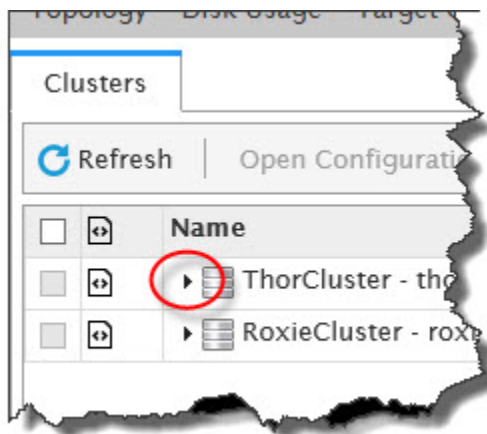
1. Click on the **Operations** icon then click on the **Cluster Processes** link.

**Figure 11. Cluster Processes Link**



2. Expand the Thor cluster by clicking on the arrow next to the **ThorCluster** link.

**Figure 12. Thor Cluster link**



3. Check the box next to any individual nodes to examine or check the **Select All** checkbox in the first row.
4. With the systems selected, the preflight action button activates and you can press it to display the preflight options.
5. Select or de-select any desired options, then press the **Submit** button at the bottom to start preflight.

Figure 13. Submit

The screenshot shows a 'Preflight' configuration window. On the left is a sidebar with a list of nodes. The main area contains various configuration options, most of which are checked. At the bottom right, the 'Submit' button is circled in red, with a mouse cursor hovering over it.

**Action:** Machine Information

**Processor Information:** ☒

**Storage Information:** ☒

**Local File Systems Only:** ☒

**Get Software Information:** ☒

**Show Processes Using Filter:** ☒

**Additional Processes To Filter:** Any Additional Processes To

**Auto Refresh:** ☒

**Auto Refresh Increment:** 5

**Warn if CPU usage is over:** 95

**Warn if available memory is under:** 95

**Warn if available disk space is under:** 95

**Submit**

node160101 Linux

## EXPECTED RESULTS:

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

Figure 14. Cluster Process results

The screenshot shows the 'Cluster Processes' tab in the software interface. It displays a table with preflight results for three nodes. The table has columns for Location, Component, Condition, State, Processes Down, Computer Up Time, and Physical Memory.

| Location                                      | Component             | Condition | State | Processes Down | Computer Up Time | Physical Mem |
|-----------------------------------------------|-----------------------|-----------|-------|----------------|------------------|--------------|
| 10.179.140.1 /var/lib/HPCCSystems/thor_160/   | Thor Slave[thor_160]  | Normal    | Ready |                | 91 days, 2:22    | 23%          |
| 10.179.140.2 /var/lib/HPCCSystems/thor_160/   | Thor Slave[thor_160]  | Normal    | Ready |                | 225 days, 3:54   | 20%          |
| 10.179.140.103 /var/lib/HPCCSystems/thor_160/ | Thor Master[thor_160] | Normal    | Ready |                | 434 days, 12:06  | 2%           |

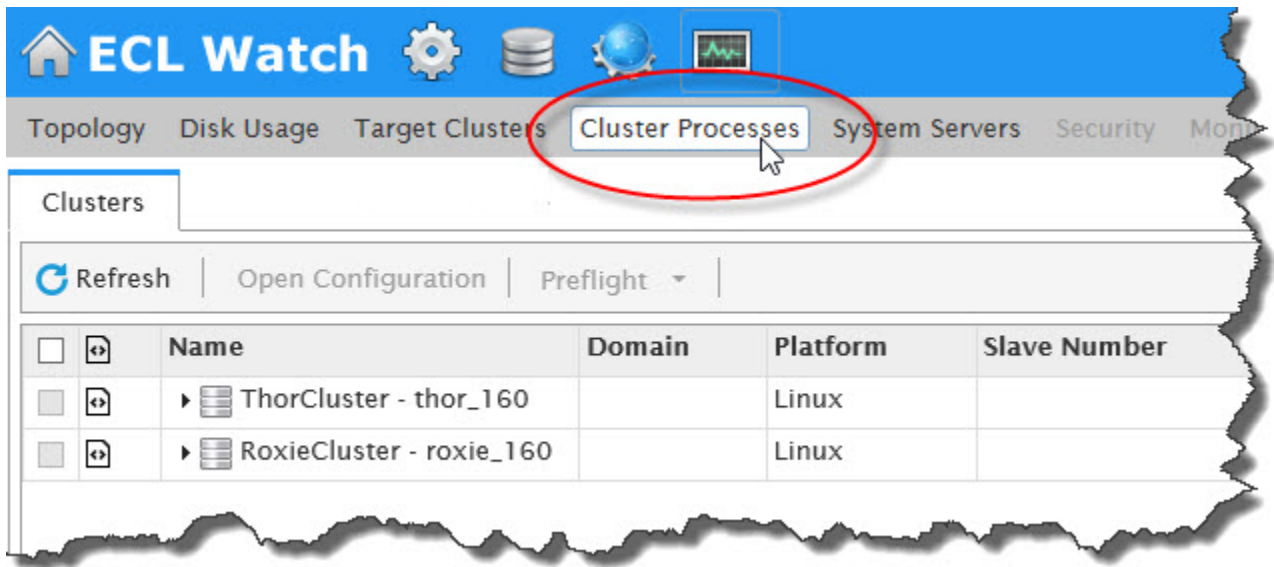
This displays information on your selected cluster(s). This information can help to indicate if everything is operating normally or can help to point out any potential concerns.

If there are any notable alerts, they are highlighted. The alerts usually require some additional attention.

# Preflight the Roxie Cluster

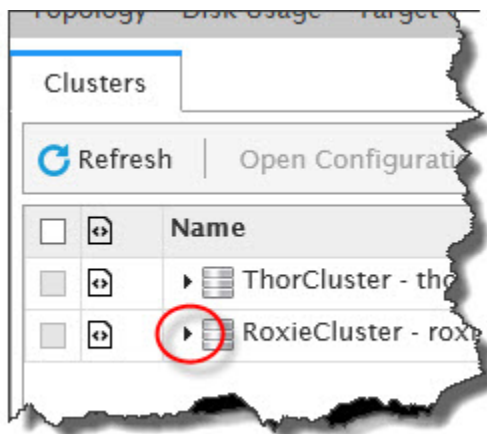
1. Click on the **Operations** icon then click on the **Cluster Processes** link.

**Figure 15. Cluster Processes Link**



2. Expand the Roxie cluster by clicking on the arrow next to the **RoxieCluster** link.

**Figure 16. RoxieCluster link**



3. Check the box next to any individual nodes to examine or check the **Select All** checkbox in the first row.
4. With the systems selected, the preflight action button activates and you can press it to display the preflight options.
5. Select or de-select any desired options, then press the **Submit** button at the bottom to start preflight.

Figure 17. Submit

The screenshot shows a 'Preflight' configuration window. The 'Action:' dropdown is set to 'Machine Information'. The following options are checked: Processor Information, Storage Information, Local File Systems Only, Get Software Information, and Show Processes Using Filter. The 'Additional Processes To Filter:' field contains 'Any Additional Processes To'. The 'Auto Refresh:' checkbox is checked. The 'Auto Refresh Increment:' is set to 5. The 'Warn if CPU usage is over:' is set to 95. The 'Warn if available memory is under:' is set to 95. The 'Warn if available disk space is under:' is set to 95. The 'Submit' button is circled in red.

## EXPECTED RESULTS

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

Figure 18. Roxie system information

The screenshot shows a table with the following data:

| Location                                     | Component                | Computer Up Time | Physical |
|----------------------------------------------|--------------------------|------------------|----------|
| 10.178.160.1 /var/lib/HPCCSystems/roxie_160/ | Roxie Server[node160001] | 4 days, 20:05    |          |
| 10.178.160.2 /var/lib/HPCCSystems/roxie_160/ | Roxie Server[node160002] | 138 days, 21:37  | 31%      |
| 10.178.160.3 /var/lib/HPCCSystems/roxie_160/ | Roxie Server[node160003] | 138 days, 21:37  | 31%      |
| 10.178.160.4 /var/lib/HPCCSystems/roxie_160/ | Roxie Server[node160004] | 138 days, 21:37  | 31%      |

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

If there are any notable alerts, they are highlighted. The alerts usually require some additional attention.

# 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

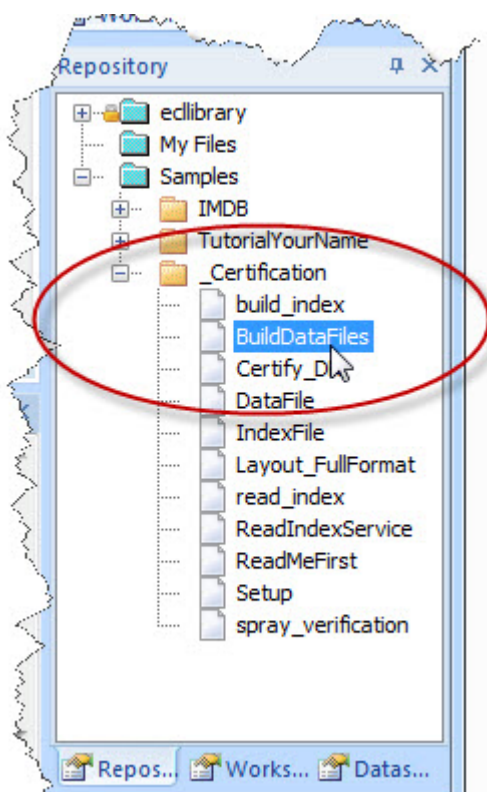
Enter the Login ID and Password provided.

|          |          |
|----------|----------|
| 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 19. ECL Files in `_Certification`**



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

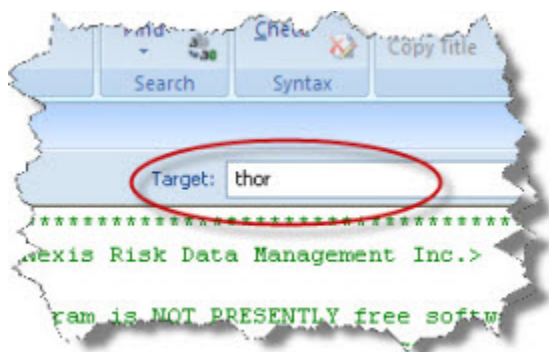
In the BuildDataFiles file, you will see some ECL code in the file as follows:

```
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.

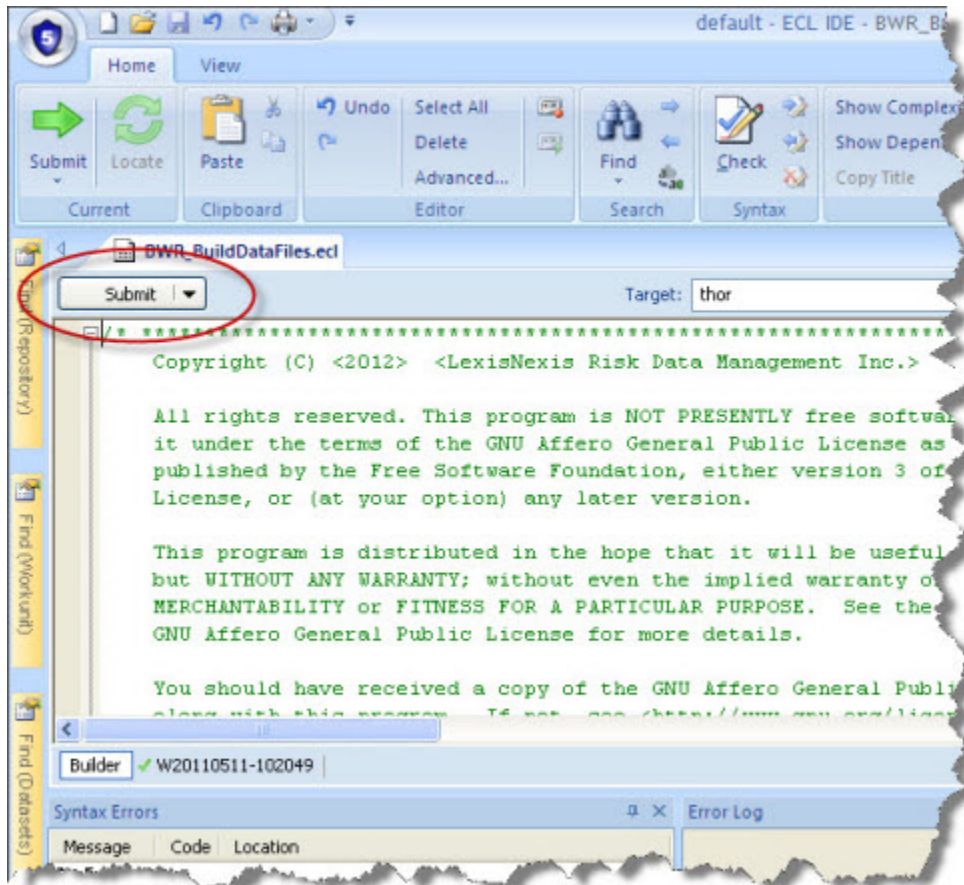
3. Select **thor** as Target from the drop menu on the right side.

**Figure 20. Target thor**



4. Press the **Submit** button.

**Figure 21. Submit button**

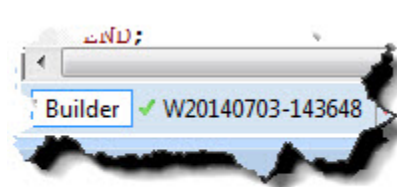


**Note:** This code generates a data file and writes it to disk.

## EXPECTED RESULT:

Look for the green checkmark indicating successful completion

**Figure 22. 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**

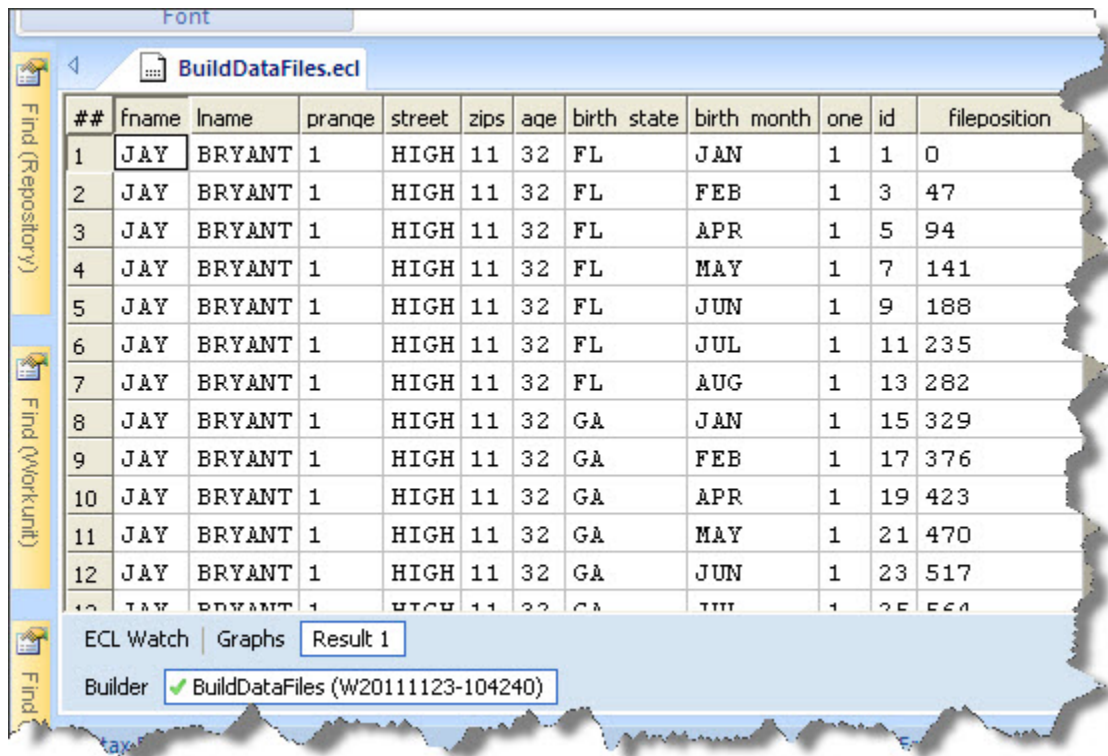
**Note:** The filename and other variables are defined in the **\_Certification.Setup** file.

**Figure 23. Workunit details page**



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

**Figure 24. View the Result**



| #  | fname | lname  | prange | street | zip | 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          |
| 13 | JAY   | BRYANT | 1      | HIGH   | 11  | 32  | GA          | JUL         | 1   | 25 | 564          |

ECL Watch | Graphs | **Result 1**

Builder ☒ BuildDataFiles (W20111123-104240)

# 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**

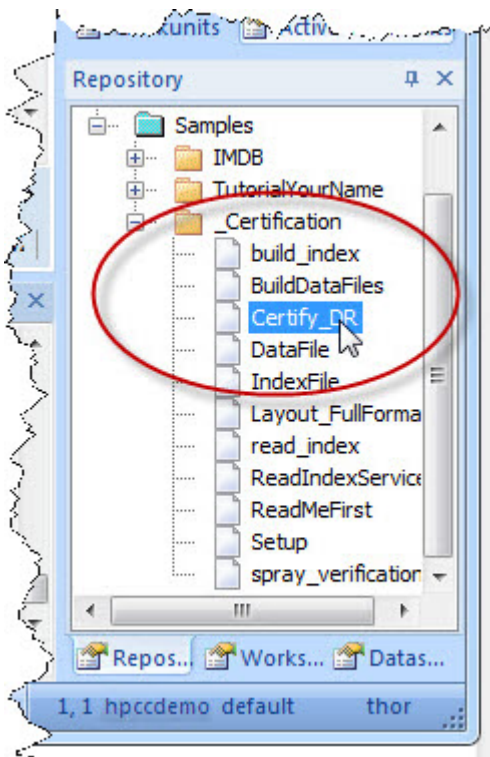
**Enter the Login ID and Password.**

|          |          |
|----------|----------|
| 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.

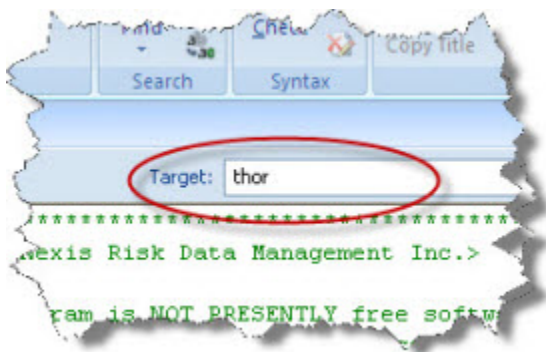
**Figure 25. ECL Certify\_DR File**



- Double-click on the **Certify\_DR** file to open it.

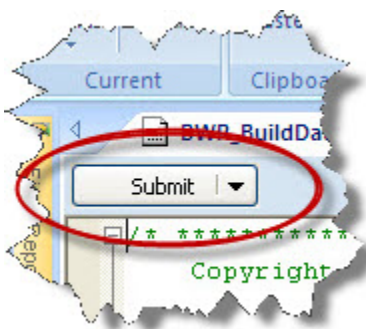
3. Select **thor** as the Target from the drop menu on the right side.

**Figure 26. Target: thor**



4. Press the **Submit** button.

**Figure 27. Submit button**

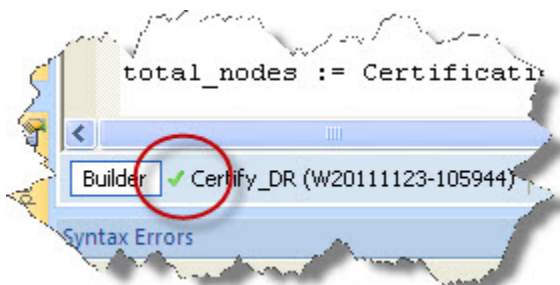


**Note:** This file uses the previously generated data and tests a series of Thor capabilities. It does not write data to disk. It will take a few minutes to complete, depending on the size of your system.

## EXPECTED RESULT

Look for the Green checkmark indicating successful completion.

**Figure 28. 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

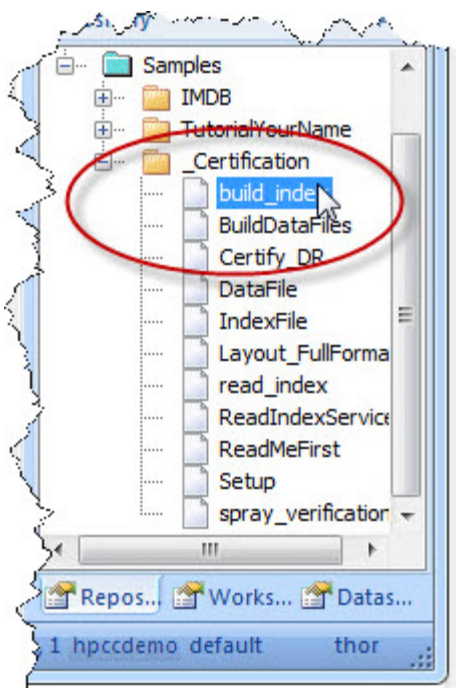
**Enter the Login ID and Password.**

|          |          |
|----------|----------|
| 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.

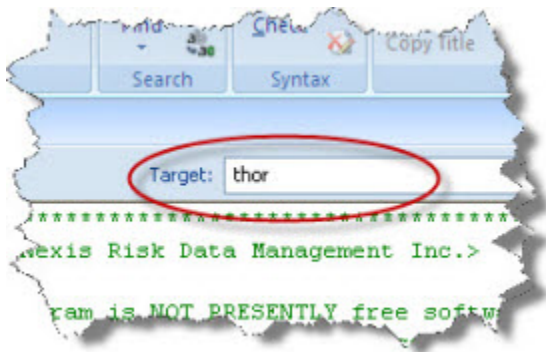
**Figure 29. Expand the `_Certification` folder**



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

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

**Figure 30. Target: thor**



4. Press the **Submit** button.

**Figure 31. Submit button**

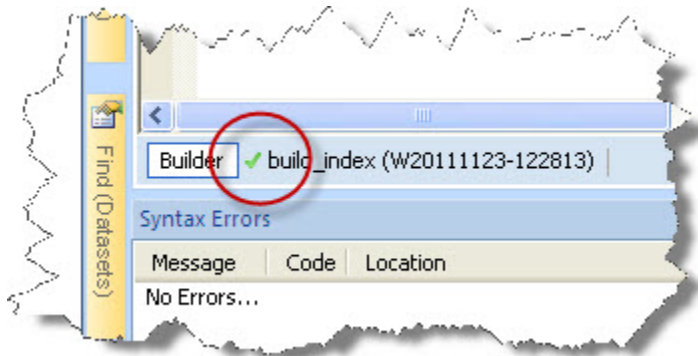


**Note:** This file uses the previously generated data. It builds an index on one of those data files.

## EXPECTED RESULT

Look for the green checkmark indicating successful completion.

**Figure 32. Green checkmark**



The file we created earlier is indexed by Last Name and the index file, **thor::full\_test\_distributed\_index**, is written to disk.

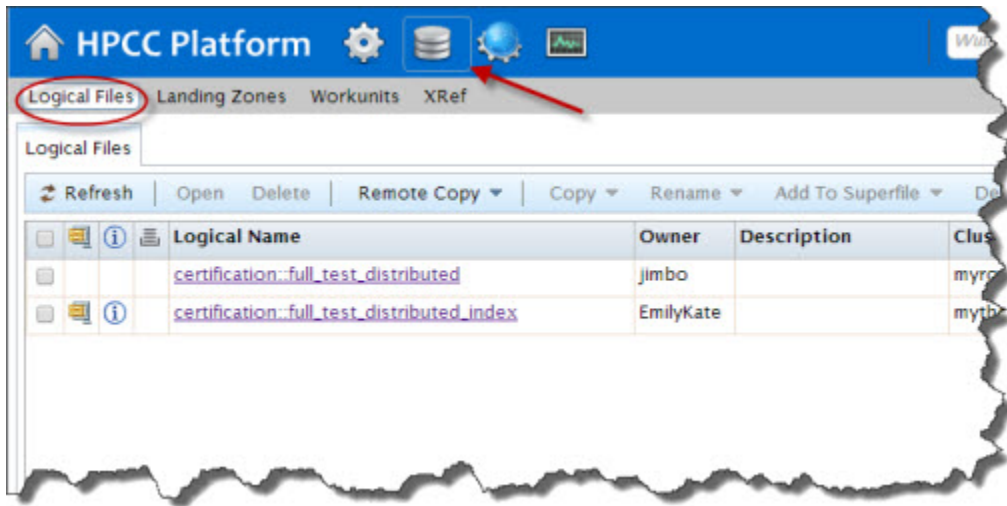
# Verify the Index Build

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 33. Browse Logical Files link**



3. Check the box next to **certification::full\_test\_distributed\_index** , then press the **Open** action button.

4. Select the Contents tab.

Figure 34. Contents

Logical Files   Landing Zones   Workunits   XRef

Logical Files   certification::full\_test\_distributed\_index   x

Summary   Contents   ECL   DEF   XML   File Parts   Queries   Workunit

Download: Zip   GZip   XLS   Filter

| ## | 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

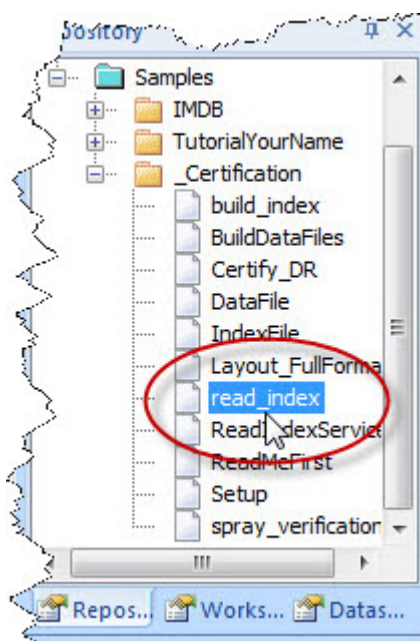
This section certifies Thor access to indexed data.

## Certify Thor Access

1. Open the `_Certification.read_index` file.

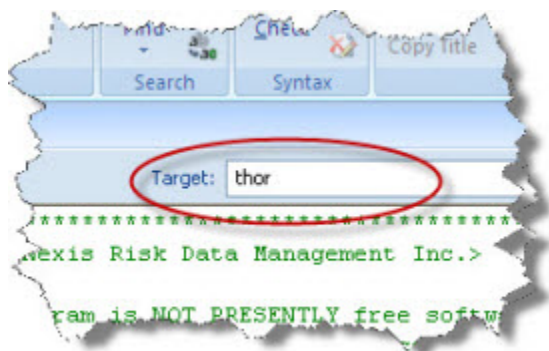
- In the lower right corner of the ECL IDE you will see a section labeled as Repository, containing a couple of folders. This contains the ECL files. Click the + sign next to *Samples*, open the folder.
- Navigate to the `_Certification` folder and click the + sign next to it to open it and view the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

**Figure 35. Expand the `_Certification` folder**



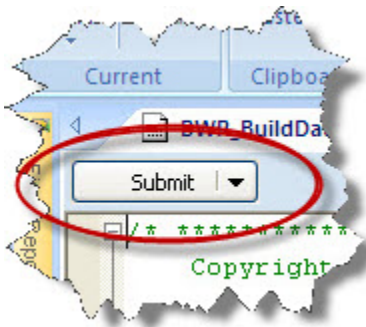
- Double-click on the `read_index` file to open it.
2. Select **thor** as Target from the drop menu on the right side.

**Figure 36. Target: thor**



3. Press the **Submit** button.

**Figure 37. Submit button**



## EXPECTED RESULT:

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

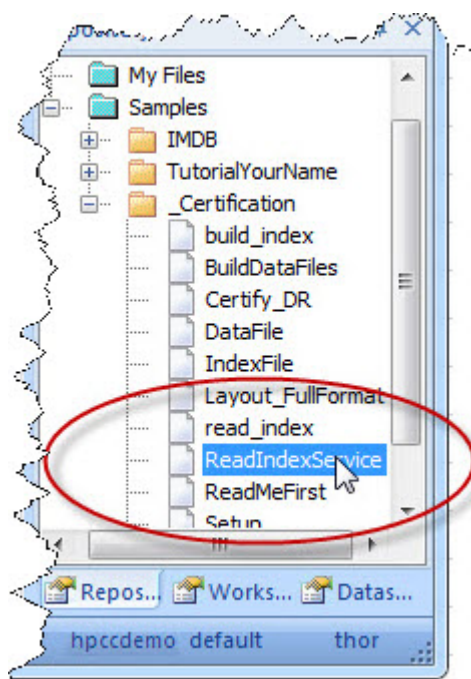
**Figure 38. Results page**

| #  | 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         |

# 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 39. \_Certification folder**



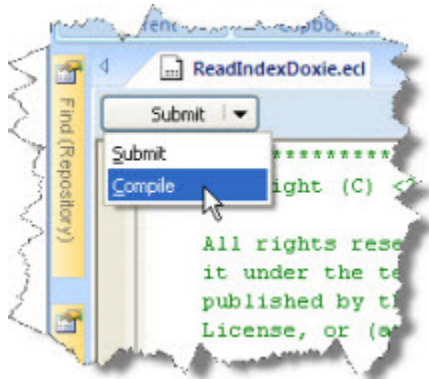
- Double-click on the **ReadIndexService** file to open it.
2. Select **roxie** as Target from the drop menu on the right side.

**Figure 40. Target roxie**



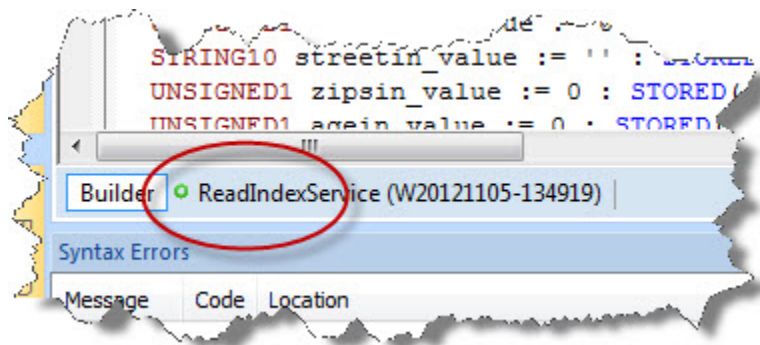
3. In the upper left corner the **Submit** button has an arrow next to it. Select the arrow to expose the **Compile** option.
  - Select **Compile** from the drop list under the submit button.

**Figure 41. Compile**



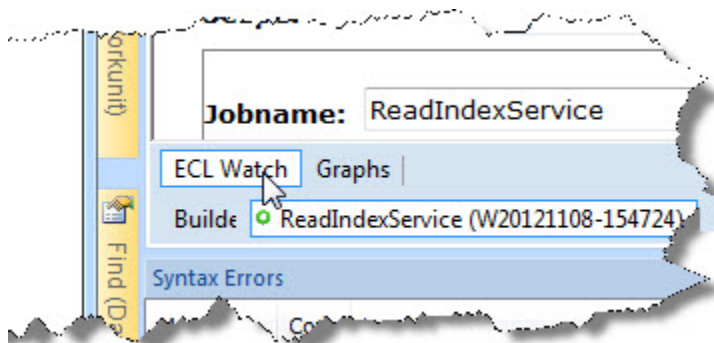
Look for a green circle indicating successful completion. Once complete, select the Workunit next to the green circle.

**Figure 42. Completed Workunit: Green circle**



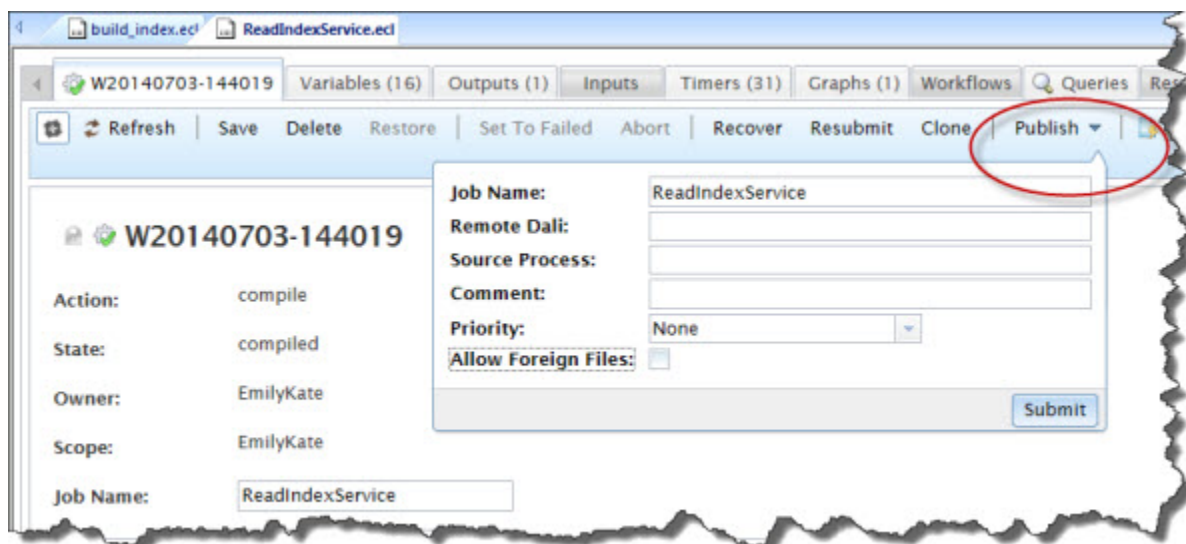
4. Select the **ECL Watch** button at the lower left corner of the window.

**Figure 43. Select ECL Watch**



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

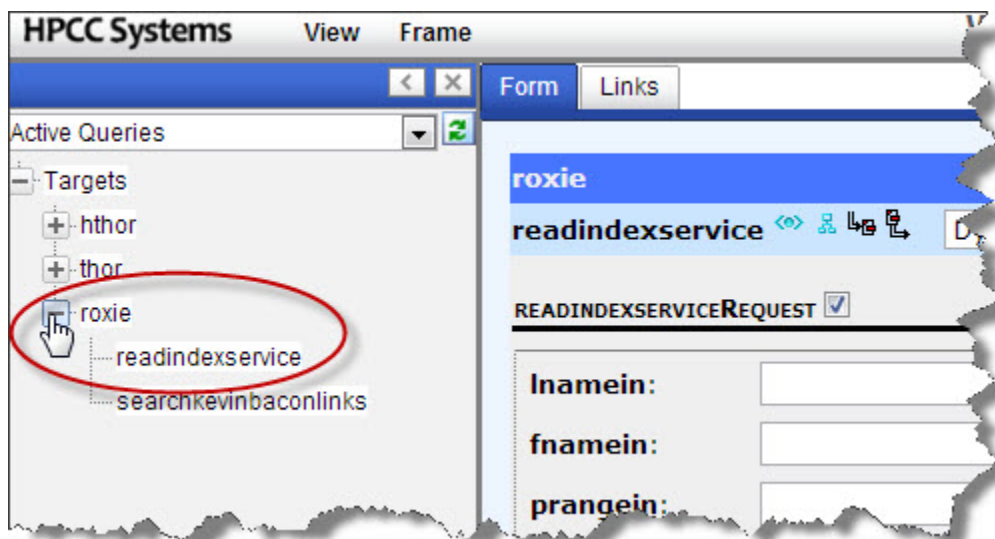
Figure 44. ECL Watch Publish button



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 45. Roxie ESP



7. Click the + sign next to **roxie**, to expand it
8. Click **readindexservice**
9. Enter the name BRYANT in the **Iname** field.

**Figure 46. Enter Inamein**

The screenshot shows a web application interface with a 'Form' tab selected. The application is titled 'roxie'. Below the title, there's a section for 'readindexservice' with a 'Dynamic Form' dropdown. A checkbox labeled 'READINDEXSERVICEREQUEST' is checked. The form contains several input fields: 'Inamein' (containing 'BRYANT'), 'fnamein', 'prangein', 'streetin', 'zipsin', 'agein', 'birth\_statein', and 'birth\_monthin'. At the bottom, there's a 'Submit' button being clicked by a mouse cursor, and a 'Clear All' button. An 'OUTPUT TABLES' dropdown is also visible.

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 47. Result

HPCC Systems

View

Frame

User

WsEC

QuerySet Aliases

QuerySets

thor

read\_index

myroxie

Form

Links

read index Response

Dataset: Result 1

|    | fname | lname  | prange | street | zip   | age | birth state | birth month | one | id      | filepos  |
|----|-------|--------|--------|--------|-------|-----|-------------|-------------|-----|---------|----------|
| 1  | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | AL          | APR         | 1   | 1569282 | 83878080 |
| 2  | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | AL          | AUG         | 1   | 1569290 | 83878268 |
| 3  | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | AL          | FEB         | 1   | 1569280 | 83878033 |
| 4  | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | AL          | JAN         | 1   | 1569278 | 83877986 |
| 5  | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | AL          | JUL         | 1   | 1569288 | 83878221 |
| 6  | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | AL          | JUN         | 1   | 1569286 | 83878174 |
| 7  | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | AL          | MAR         | 1   | 1568113 | 36850632 |
| 8  | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | AL          | MAY         | 1   | 1569284 | 83878127 |
| 9  | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | AL          | OCT         | 1   | 1568117 | 36850726 |
| 10 | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | AL          | SEP         | 1   | 1568115 | 36850679 |
| 11 | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | CA          | APR         | 1   | 881965  | 20726154 |
| 12 | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | CA          | AUG         | 1   | 881973  | 20726342 |
| 13 | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | CA          | FEB         | 1   | 881963  | 20726107 |
| 14 | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | CA          | JAN         | 1   | 881961  | 20726060 |
| 15 | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | CA          | JUL         | 1   | 881971  | 20726295 |
| 16 | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | CA          | JUN         | 1   | 881969  | 20726248 |
| 17 | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | CA          | MAR         | 1   | 381110  | 55956038 |
| 18 | DIRK  | BRYANT | 1      | 25TH   | 11 31 | 31  | CA          | MAY         | 1   | 881967  | 20726001 |

# 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)**

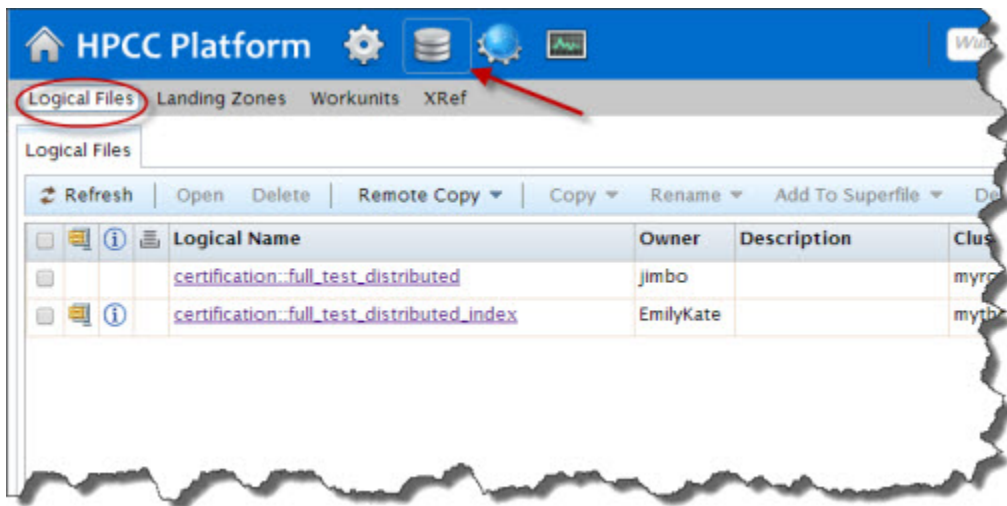


**NOTE:** To copy a URL to the clipboard, click the  icon from the row of icons along the top of the ECL Watch page.

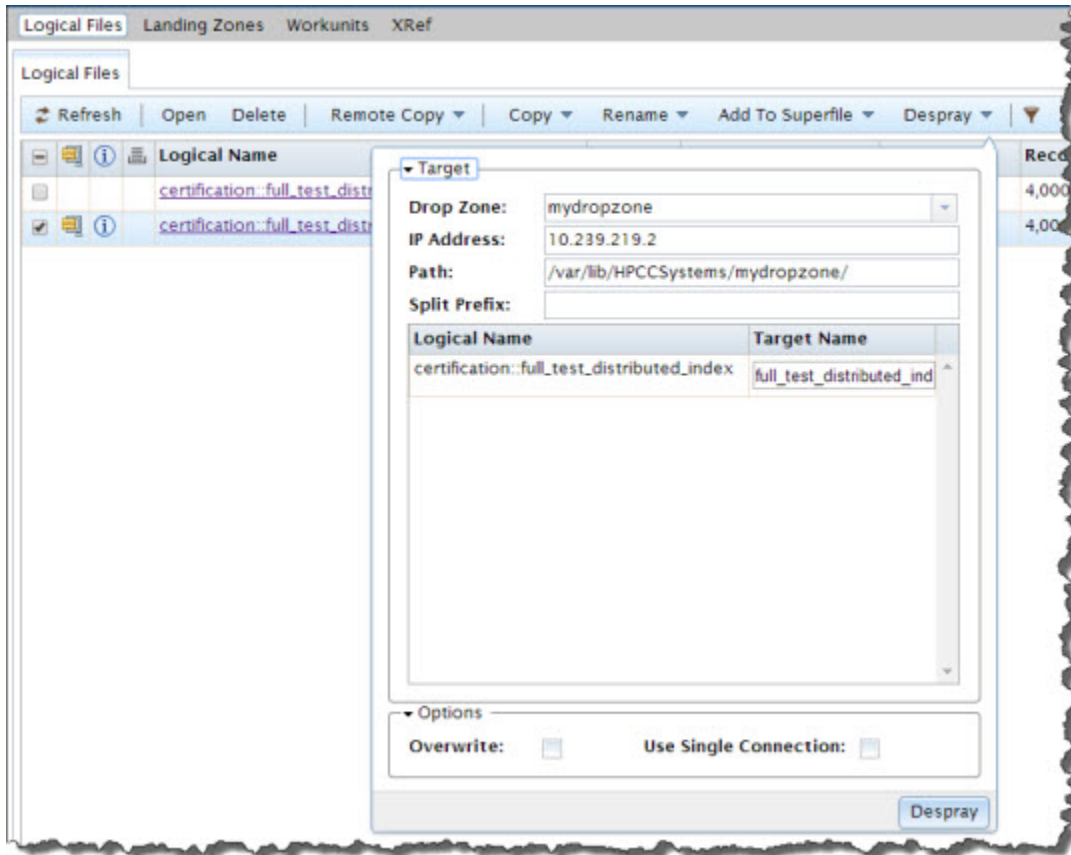
This opens a window with the full URL. Select the entire URL and you can copy it to the clipboard to paste elsewhere.

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

**Figure 48. Browse Files**



3. Check the box next to **certification::full\_test\_distributed\_index**, then press the **Despray** action button.



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 configuration. 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 file-name, 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 Systems 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 Distributed 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.



4. Fill in relevant details:

#### Target

- Group** Select the name of cluster to spray to. You can only select a cluster in your environment.
- Queue** Select the queue for the spray.
- Target Scope** The prefix for the logical file, in this case **certification**

|                               |                                                                                                                                                                                                                                                          |
|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Target Name</b>            | The logical filename to create. This is pre-filled with the name of the source file on the landing zone, but can be changed.                                                                                                                             |
| <b>Record Length</b>          | The size of each record. In this case it is <b>47</b>                                                                                                                                                                                                    |
| <b>Options:</b>               |                                                                                                                                                                                                                                                          |
| <b>Overwrite</b>              | Check this box to overwrite files of the same name.                                                                                                                                                                                                      |
| <b>Replicate</b>              | 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).<br><br><b>This option is only available on systems where replication has been enabled.</b> |
| <b>Compress</b>               | Check this box to compress the files.                                                                                                                                                                                                                    |
| <b>No Split</b>               | Check this box to prevent splitting file parts to multiple target parts.                                                                                                                                                                                 |
| <b>Expire in (days)</b>       | An integer value indicating the number of days before automatically removing the file. If omitted, the default is -1 (never expires).                                                                                                                    |
| <b>Fail if no source file</b> | Check this box to allow the spray to fail if no source file is found.                                                                                                                                                                                    |

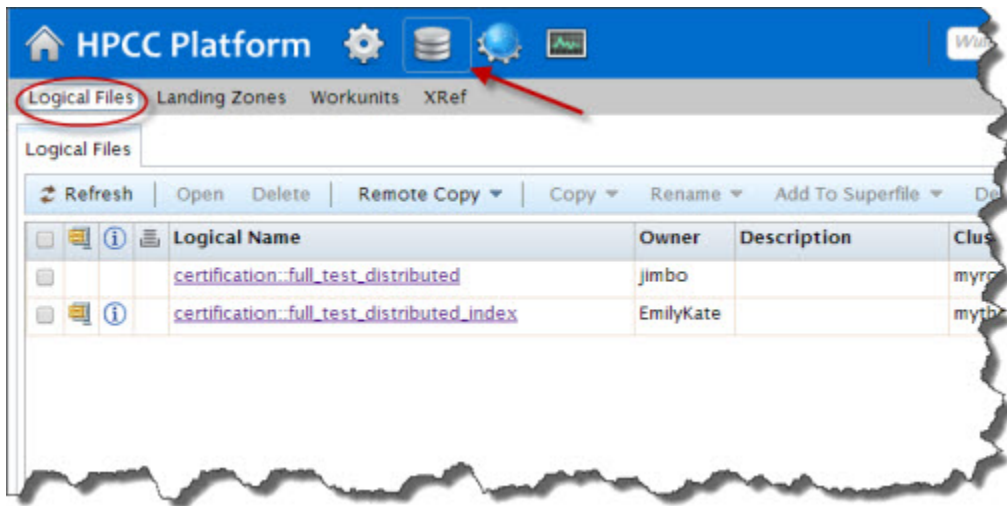
5. Press the **Spray** button.

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 49. Browse Files**



2. Click on the sprayed file, select **Open** to view the logical file details..
3. Select the Contents tab to view contents.