

IBM Spectrum Conductor with Spark V2.2.0 RFE 97658

Readme File

About RFE 97658

This package contains the Conductor log path/directory structure enhancement for the human process feature, which enables a new logging structure that can be uniquely identified. Use this feature when you want to enhance the Conductor log path structure. Benefits of using this feature include, organizing all Spark instance group logs under one directory with human readable names and structure. Without this feature, the Conductor Log is organized for the Elastic Stack and is hard for humans to find/search the logs.

Readme file for: IBM® Spectrum Conductor with Spark

Product/Component Release: 2.2.0

Fix ID: cws-2.2-build469470-jpmc

Publication date: 29 September 2017

Last modified date: 29 September 2017

1.	Scope	3
2.	Configuration to enable RFE 97658.....	3
1)	Prerequisites.....	3
2)	Installation files	3
3)	Installation procedure	3
4)	Configuration procedure	7
3.	How this feature works.....	9
1)	Current IBM Spectrum Conductor with Spark log structure	9
2)	IBM Spectrum Conductor with Spark log structure after this feature is enabled	9
4.	Copyright and trademark information.....	10

1. Scope

Applicability	
Operating system	Management and compute hosts: <ul style="list-style-type: none">Linux 64-bit
Product version	IBM Spectrum Conductor with Spark 2.2.0
Spark version	2.1.0

2. Configuration to enable RFE 97658

1) Prerequisites

Before you begin, IBM Spectrum Conductor with Spark V2.2.0 must be installed on a supported operating system. For more information, see

https://www.ibm.com/support/knowledgecenter/SSZU2E_2.2.0/installing/install_upgrade.html.

2) Installation files

This package includes the following files:

File name	Description
Spark2.1.0-Conductor2.2.0.tgz	New Spark version 2.1.0 package
lifecycle.tgz	Patch package

3) Installation procedure

Before installation

- Log on to each management host and backup the following files:

```
> mkdir -p /tmp/backup-469470
> cp $EGO_TOP/3.5/linux-x86_64/lib/librfa.so /tmp/backup-469470
> cp $EGO_TOP/ascd/2.2.0/lib/asc-common-2.2.0.jar /tmp/backup-469470
> cp $EGO_TOP/ascd/2.2.0/lib/asc-core-2.2.0.jar /tmp/backup-469470
> cp $EGO_TOP/conductorspark/2.2.0/etc/cleanup.py /tmp/backup-469470
> cp $EGO_TOP/conductorspark/2.2.0/etc/recovery.py /tmp/backup-
```

469470

```
> cp $EGO_TOP/gui/3.5/lib/egogui.jar /tmp/backup-469470
> cp $EGO_TOP/integration/elk/1.2/scripts/startshipper.sh
/tmp/backup-469470
> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/appsandnotebooks/applicationView.jsp /tmp/backup-469470
> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/appsandnotebooks/appsAndNotebooksList.jsp /tmp/backup-469470
> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/appsandnotebooks/sharedApplications.html /tmp/backup-469470
> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/appsandnotebooks/submittedApplications.html /tmp/backup-469470
> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/appsandnotebooks/js/applicationDriversAndExecutors.controller.js
/tmp/backup-469470
> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/appsandnotebooks/js/applicationView.controller.js /tmp/backup-
469470
> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/appsandnotebooks/js/appsAndNotebooksList.controller.js /tmp/backup-
469470
> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/appsandnotebooks/js/sharedApplications.controller.js /tmp/backup-
469470
> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/appsandnotebooks/js/viewScheduleHistory.controller.js /tmp/backup-
469470
> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/common/js/ConductorSparkApp.js /tmp/backup-469470
> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/instance/instanceView.jsp /tmp/backup-469470
> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/instance/js/instances.controller.js /tmp/backup-469470
> cp
```

```

$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/instance/js/instanceViewApplications.controller.js /tmp/backup-
469470

> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/conductorgui/spark
/instance/js/instanceView.controller.js /tmp/backup-469470

> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/cwsguiv5/appsandno
tebooks/js/appsandnotebooks.controller.js /tmp/backup-469470

> cp
$EGO_TOP/wlp/usr/servers/gui/apps/conductor/2.2.0/cwsguiv5/instance/
js/sigView.controller.js /tmp/backup-469470

> cp
$EGO_TOP/wlp/usr/servers/gui/apps/ego/3.5/platform/common/serviceIns
tanceLogViewPreference.jsp /tmp/backup-469470

> cp
$EGO_TOP/wlp/usr/servers/gui/apps/ego/3.5/platform/logretrieve/servi
ceInstanceLog.jsp /tmp/backup-469470

> cp
$EGO_TOP/wlp/usr/servers/gui/apps/ego/3.5/platform/logretrieve/showT
able.jsp /tmp/backup-469470

> cp $EGO_CONFDIR/../../integration/elk/conf/indexer/spark.conf
/tmp/backup-469470

> cp $EGO_CONFDIR/../../integration/elk/conf/shipper/conductor.yml
/tmp/backup-469470

```

2. Log on to each compute host and backup the following files:

```

> mkdir -p /tmp/backup-469470-comp
> cp $EGO_TOP/3.5/linux-x86_64/lib/librfa.so /tmp/backup-469470-comp
> cp $EGO_TOP/conductorspark/2.2.0/etc/cleanup.py /tmp/backup-
469470-comp
> cp $EGO_TOP/conductorspark/2.2.0/etc/recovery.py /tmp/backup-
469470-comp
> cp $EGO_TOP/integration/elk/1.2/scripts/startshipper.sh
/tmp/backup-469470-comp
> cp $EGO_CONFDIR/../../integration/elk/conf/shipper/conductor.yml
/tmp/backup-469470-comp

```

Installation steps

1. Log on to the master host as the cluster administrator:


```
> egosh user logon -u Admin -x Admin
```
2. Stop the ascd, WEBGUI, SparkCleanup, elk-indexer, and elk-shipper services:


```
> egosh service stop ascd
> egosh service stop WEBGUI
```

```
> egosh service stop SparkCleanup
> egosh service stop elk-indexer
> egosh service stop elk-shipper
```

3. Log on to each management host in your cluster as the cluster administrator and decompress the **cws-2.2.0.0_x86_64_build469470.tgz** package:

```
> mkdir -p /tmp/build469470
> tar -zxof cws-2.2.0.0_x86_64_build469470.tgz -C /tmp/build469470
> tar -zxof /tmp/build469470/lifecycle.tgz -C $EGO_TOP
```

4. Log on to each compute host in your cluster as the cluster administrator and decompress the **cws-2.2.0.0_x86_64_build469470.tgz** package:

```
> mkdir -p /tmp/build469470-comp
> tar -zxof cws-2.2.0.0_x86_64_build469470.tgz -C /tmp/build469470-comp
> tar -zxof /tmp/build469470-comp/lifecycle.tgz -C /tmp/build469470-comp
> cp /tmp/build469470-comp/3.5/linux-x86_64/lib/librfa.so
$EGO_TOP/3.5/linux-x86_64/lib/librfa.so
> cp /tmp/build469470-comp/conductorspark/2.2.0/etc/cleanup.py
$EGO_TOP/conductorspark/2.2.0/etc/cleanup.py
> cp /tmp/build469470-comp/conductorspark/2.2.0/etc/recovery.py
$EGO_TOP/conductorspark/2.2.0/etc/recovery.py
> cp /tmp/build469470-comp/integration/elk/1.2/scripts/startshipper.sh
$EGO_TOP/integration/elk/1.2/scripts/startshipper.sh
> cp /tmp/build469470-comp/integration/elk/conf/shipper/conductor.yml
$EGO_TOP/integration/elk/conf/shipper/conductor.yml
```

5. If you enabled master host failover in your cluster, log on to the master host as the cluster administrator and run the following commands:

```
> cp $EGO_TOP/integration/elk/conf/indexer/spark.conf
$EGO_CONFDIR/../../integration/elk/conf/indexer/spark.conf
> cp $EGO_TOP/integration/elk/conf/shipper/conductor.yml
$EGO_CONFDIR/../../integration/elk/conf/shipper/conductor.yml
```

6. Verify that permissions and ownership of the replaced files are the same as they were before applying the fix. Update any file permissions or ownership as required.

7. Delete all subdirectories and files in the `$EGO_TOP/ascd/workarea` directory on the management hosts:

```
> rm -rf $EGO_TOP/ascd/workarea/*
```

NOTE: If you changed the default configuration for the `WLP_OUTPUT_DIR` environment variable and `APPEND_HOSTNAME_TO_WLP_OUTPUT_DIR` is set to `true` in the `$EGO_CONFDIR/wlp.conf` file, you must clean up the `$WLP_OUTPUT_DIR/ascd_hostname/ascd/workarea/` directory.

8. Delete all subdirectories and files in the GUI work directory on the management hosts:

```
> rm -rf $EGO_TOP/gui/work/*
> rm -rf $EGO_TOP/gui/workarea/*
```

NOTE: If you changed the default configuration for the `WLP_OUTPUT_DIR` environment variable and `APPEND_HOSTNAME_TO_WLP_OUTPUT_DIR` is set to `true` in the `$EGO_CONFDIR/wlp.conf` file, you must clean up the `$WLP_OUTPUT_DIR/webgui_hostname/gui/workarea/` directory.

9. Launch your browser and clear the browser cache.

After installation

1. Start the `ascd`, `WEBGUI`, `SparkCleanup`, `elk-indexer`, and `elk-shipper` services:

```
> egosh service start ascd
> egosh service start WEBGUI
> egosh service start SparkCleanup
> egosh service start elk-indexer
> egosh service start elk-shipper
```

2. On the client machine where you have a browser, decompress the **cws-2.2.0.0_x86_64_build469470.tgz** package:

```
> mkdir -p /tmp/build469470
> tar zxof cws-2.2.0.0_x86_64_build469470.tgz -C /tmp/build469470
```

3. Launch a browser and clear the browser cache. Login to the cluster management console as the cluster administrator.
4. Add the Spark version 2.1.0 package to your cluster:
 - a. Click **Workload > Spark > Version Management > Add**.
 - b. Click **Browse** and select the `/tmp/build469470/Spark2.1.0-Conductor2.2.0.tgz` package.
 - c. Click **Add**.

4) Configuration procedure

To enable this feature:

1. Stop the `ascd`, `SparkCleanup`, `elk-indexer`, and `elk-shipper` services:

```
> egosh service stop ascd
> egosh service stop SparkCleanup
> egosh service stop elk-indexer
> egosh service stop elk-shipper
```

2. Log on to the management hosts, add **SPARK_EGO_LOCAL_LOGGING=2.1.0** in `$EGO_CONFDIR/../../ascd/conf/ascd.conf` (you need to run once only if `$EGO_CONFDIR` is a shared directory).

3. Log on to each host, create the ELASTIC_HARVEST_LOCATION directory and set permission to 777. You need to run once only if ELASTIC_HARVEST_LOCATION is a shared directory. For example:

```
> mkdir ${EGO_TOP}/elk_logs  
> chmod 777 ${EGO_TOP}/elk_logs
```

NOTE: ELASTIC_HARVEST_LOCATION must be a shared directory if IBM Spectrum Conductor with Spark cluster is installed to a shared environment, and the value must be different from ELK_HARVEST_LOCATION.

4. Launch a web browser and clear the browser cache. Log in to the cluster management console as the cluster administrator.
 - a. Click **System & Services > EGO Services > Service Profiles**.
 - b. Open the **elk-shipper** service profile in the **System Services** tab.
 - c. In the Actions list of the **ego:ActivitySpecification** section, click **Insert “ego:EnvironmentVariable”**, set the name to **ELASTIC_HARVEST_LOCATION**, and then set the content to the directory created in step 3. For example:
`${EGO_TOP}/elk_logs.`
 - d. Change the value of **ego:ExecutionUser** to **root** in the **ego:ActivitySpecification** section.
 - e. Click **Save** button.
 - f. Open the **SparkCleanup** service profile in the **Other Services** tab.
 - g. In the Actions list of the **ego:ActivitySpecification** section, click **Insert “ego:EnvironmentVariable”**, set the name to **ELASTIC_HARVEST_LOCATION**, and then set the content to the directory created in step 3. For example:
`${EGO_TOP}/elk_logs.`
 - h. Click **Save** button.
5. Start the ascd, SparkCleanup, elk-indexer, and elk-shipper services:

```
> egosh service start ascd  
> egosh service start SparkCleanup  
> egosh service start elk-indexer  
> egosh service start elk-shipper
```
6. From the cluster management console, click **Workload > Spark > Spark Instance Groups**.
 - a. Create a new Spark instance group that uses Spark 2.1.0. For details, see [Creating Spark instance groups](#).
 - b. If required, upgrade your existing Spark instance groups that uses Spark 2.1.0. For details, see [Updating existing Spark instance groups](#).

To disable this feature:

1. Log on to the management hosts and remove **SPARK_EGO_LOCAL_LOGGING=2.1.0** in `${EGO_CONFDIR}/../..../ascd/conf/ascd.conf`. You need to run once only if `${EGO_CONFDIR}` is a shared directory.

3. How this feature works

1) Current IBM Spectrum Conductor with Spark log structure

Spark driver and executor logs:

```
$ELK_HARVEST_LOCATION/<uuid>.<egouser>.<signame>.<osexecuser>
|-- diver-<uuid>
    |-- stdout
    |-- stderr
|-- app-<uuid>
    |-- <uuid>
        |-- stdout
        |-- stderr
```

Conductor log (Spark master event):

```
$ELK_HARVEST_LOCATION/<uuid>.<signame>
|-- conductor.<uuid>.<signame>.<egosc_service_name>.<ego_instance_seqno>.<date>
|-- conductor.<uuid>.<signame>.<egosc_service_name>.<ego_instance_seqno>.recovery
```

These logs are currently organized for Elastic Stack and are hard for human to find/search the logs.

2) IBM Spectrum Conductor with Spark log structure after this feature is enabled

Highlighted differences/enhancements:

- The top log's directory for each Spark instance group is now placed under the Spark instance group deployment location: **\$SPARK_HOME/logs**.
- Symbolic link **\$ELASTIC_HARVEST_LOCATION/<uuid>.signame** → per Spark instance group's **\$SPARK_HOME/logs**.
- **ELASTIC_HARVEST_LOCATION** is the new Elastic Stack harvest location for new logging. If IBM Spectrum Conductor with Spark is installed to a shared environment, the location should be on shared file system.
- Renamed **driver** directory to include **users and hostname**.
- Renamed **app** directory to include **users**.
- Renamed **executors** directory to include **hostname and incremental integer** (start from 0).
- Moved and renamed **conductor** events log to include **hostname**.
- Keep overall existing log file structure under each Spark instance group's logs (**\$SPARK_HOME/logs**) directory.

```
$SPARK_HOME/logs <== symbolic link $ELASTIC_HARVEST_LOCATION/<uuid>.signame
|-- conductor.<egosc_service_name>.<ego_instance_seqno>.<host>
    //conductor log file
|-- diver-<uuid>.< egouser>.<osexecuser>.<host>
    //driver log dir
    |-- stdout
    //driver output file
    |-- stderr
    //driver error log file
|-- app-<uuid>.< egouser>.<osexecuser>
    //app log dir
    |-- <incremental_integer>-<uuid>.<host>
    //executor log dir
        |-- stdout
        //executor output file
        |-- stderr
        //executor error log file
|-- <incremental_integer>-<uuid>.<host>
    //executor log dir
        |-- stdout
        //executor output file
        |-- stderr
        //executor error log file
```

4. Copyright and trademark information

© Copyright IBM Corporation 2017

U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

IBM®, the IBM logo and ibm.com® are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.