EBook Gratis

APRENDIZAJE oozie

Free unaffiliated eBook created from
Stack Overflow contributors.

#oozie

Tabla de contenido

Acerca de

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: oozie

It is an unofficial and free oozie ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official oozie.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Capítulo 1: Empezando con oozie

Observaciones

Oozie es un proyecto de código abierto Apache, desarrollado originalmente en Yahoo. Oozie es un sistema de programación de propósito general para trabajos de Hadoop de varias etapas.

- Oozie permite formar una agrupación lógica de trabajos Hadoop relevantes en una entidad llamada `Workflow` . Los flujos de trabajo de Oozie son DAG (Gráfico cíclico dirigido) de acciones.
- Oozie proporciona una forma de programar el flujo de trabajo dependiente de la **hora** o los **datos** utilizando una entidad llamada `Coordinator` .
- Además, puede combinar los Coordinadores relacionados en una entidad llamada `Bundle` y puede programarse en un servidor Oozie para su ejecución.

Oozie admite la mayoría de los trabajos Hadoop como nodos de acción Oozie como: `MapRedude` , `Java` , `FileSystem` (operaciones HDFS), `Hive` , `Hive2` , `Pig` , `Spark` , `SSH` , `Shell` , `DistCp` y `Sqoop` . Proporciona una capacidad de decisión utilizando una acción de `Decision Control Node` decisión y la ejecución paralela de los trabajos utilizando el `Fork-Join Control Node` . Permite a los usuarios configurar la opción de correo electrónico para la notificación de éxito / falla del flujo de trabajo mediante la acción de `Email` .

Versiones

Versión de Oozie	Fecha de lanzamiento
4.3.0	2016-12-02

Examples

Instalación o configuración

Requisitos previos

Este artículo demostró la instalación de oozie-4.3.0 en Hadoop 2.7.3

1. Java 1.7+
2. Hadoop 2.x (aquí, 2.7.3)
3. Maven3 +
4. Caja de Unix

Paso 1: archivo dist

Obtenga el archivo oozie tar.gz desde http://www-eu.apache.org/dist/oozie/4.3.0/ y extráigalo

```
cd $HOME
tar -xvf oozie-4.3.0.tar.gz
```

Paso 2: Construir Oozie

```
cd $HOME/oozie-4.3.0/bin
./mkdistro.sh -DskipTests
```

Paso 3: Instalación del servidor

Copie los archivos binarios construidos en el directorio de inicio como 'oozie'

```
cd $HOME
cp -R $HOME/oozie-4.3.0/distro/target/oozie-4.3.0-distro/oozie-4.3.0 .
```

Paso 3.1: libext Crear el directorio libext dentro del directorio oozie

```
 cd $HOME/oozie
 mkdir libext
```

Nota : biblioteca ExtJS (2.2+) (opcional, para habilitar la consola web de Oozie) Sin embargo, la biblioteca ExtJS no se incluye con Oozie porque usa una licencia diferente :(Ahora debe colocar los archivos hadoop en el directorio libext, de lo contrario se lanzará a continuación error en el archivo oozie.log

> WARN ActionStartXCommand: 523 - SERVIDOR [data01.teg.io] USUARIO [hadoop] GRUPO [-] TOKEN [] APLICACIÓN [map-reduce-wf] TRABAJO [0000000-161215143751620-oozie-hado-W] ACCIÓN [0000000-161215141437516 20- oozie-hado-W @ mr-node] Error al iniciar la acción [mr-node]. ErrorType [TRANSIENT], ErrorCode [JA009], Mensaje [JA009: No se puede inicializar Cluster. Verifique su configuración para mapreduce.framework.name y las direcciones de servidor correspondientes.]

Entonces, vamos a poner debajo de los frascos dentro del directorio libext

```
cp $HADOOP_HOME/share/hadoop/common/*.jar oozie/libext/
cp $HADOOP_HOME/share/hadoop/common/lib/*.jar oozie/libext/
cp $HADOOP_HOME/share/hadoop/hdfs/*.jar oozie/libext/
cp $HADOOP_HOME/share/hadoop/hdfs/lib/*.jar oozie/libext/
cp $HADOOP_HOME/share/hadoop/mapreduce/*.jar oozie/libext/
cp $HADOOP_HOME/share/hadoop/mapreduce/lib/*.jar oozie/libext/
cp $HADOOP_HOME/share/hadoop/yarn/*.jar oozie/libext/
cp $HADOOP_HOME/share/hadoop/yarn/lib/*.jar oozie/libext/
```

Paso 3.2: Oozie Impersonate Para evitar el error de suplantación en oozie, modifique core-site.xml como a continuación

```
<!-- OOZIE -->
  <property>
    <name>hadoop.proxyuser.[OOZIE_SERVER_USER].hosts</name>
```

```
    <value>[OOZIE_SERVER_HOSTNAME]</value>
  </property>
  <property>
    <name>hadoop.proxyuser.[OOZIE_SERVER_USER].groups</name>
    <value>[USER_GROUPS_THAT_ALLOW_IMPERSONATION]</value>
  </property>
```

Asumiendo, mi usuario de oozie es huser y el host es localhost y el grupo es hadoop

```
<!-- OOZIE -->
  <property>
    <name>hadoop.proxyuser.huser.hosts</name>
    <value>localhost</value>
  </property>
  <property>
    <name>hadoop.proxyuser.huser.groups</name>
    <value>hadoop</value>
  </property>
```

Nota: Puedes usar * en todos los valores, en caso de confusión.

Paso 3.3: Preparar la guerra.

```
cd $HOME/oozie/bin
./oozie-setup.sh prepare-war
```

Esto creará el archivo oozie.war dentro del directorio oozie. Si esta guerra se usará más, puede enfrentar este error:

> ERROR ActionStartXCommand: 517 - SERVIDOR [data01.teg.io] USUARIO [hadoop] GRUPO [-] TOKEN [] APP [map-reduce-wf] TRABAJO [0000000-161220104605103-oozie-hado-W] ACCIÓN [0000000-1612201046051053- oozie-hado-W @ mr-node] Error, java.lang.NoSuchFieldError: HADOOP_CLASSPATH

¿Por qué? porque, La compilación de oozie produjo archivos Hadoop 2.6.0 incluso al especificar Hadoop 2.7.3 con la opción "-Dhadoop.version = 2.7.3".

Entonces, para evitar este error, copie el archivo oozie.war en un directorio diferente

```
mkdir $HOME/oozie_war_dir
cp $HOME/oozie/oozie.war $HOME/oozie_war_dir
cd $HOME/oozie_war_dir
jar -xvf oozie.war
rm -f oozie.war/WEB-INF/lib/hadoop-*.jar
rm -f oozie.war/WEB-INF/lib/hive-*.jar
rm oozie.war
jar -cvf oozie.war ./*
cp oozie.war $HOME/oozie/
```

Luego, regenera los binarios oozie.war para oozie con una guerra de preparación

```
cd $HOME/oozie/bin
./oozie-setup.sh prepare-war
```

Paso 3.4: Crear sharelib en HDFS

```
cd $HOME/oozie/bin
./oozie-setup.sh sharelib create -fs hdfs://localhost:9000
```

Ahora, esta configuración de sharelib puede darte un error a continuación:

> org.apache.oozie.service.ServiceException: E0104: No se pudo inicializar completamente el servicio [org.apache.oozie.service.ShareLibService], no se pudo almacenar en caché sharelib. Un administrador necesita instalar el sharelib con oozie-setup.sh y emitir el comando CLI 'oozie admin' para actualizar el sharelib

Para evitar esto, modifique oozie-site.xml como a continuación

```
cd $HOME/oozie
vi conf/oozie-site.xml
```

Añadir propiedad abajo

```
<property>
    <name>oozie.service.HadoopAccessorService.hadoop.configurations</name>
    <value>*=/usr/local/hadoop/etc/hadoop/</value>
</property>
```

El valor debe ser su $ HADOOP_HOME / etc / hadoop, donde están presentes todos los archivos de configuración de hadoop.

Paso 3.5: Crear Oozie DB

```
cd $HOME/oozie
./bin/ooziedb.sh create -sqlfile oozie.sql -run
```

Paso 3.6: Iniciar Daemon

Para iniciar Oozie como demonio use el siguiente comando:

```
./bin/oozied.sh start
```

Para detener

```
./bin/oozied.sh stop
```

verifique los registros para los errores, si los hay

```
cd $HOME/oozie/logs
tail -100f oozie.log
```

Use el siguiente comando para verificar el estado de Oozie desde la línea de comandos:

```
$ ./bin/oozie admin -oozie http://localhost:11000/oozie -status
System mode: NORMAL
```

Paso 4: Instalación del cliente

```
$ cd
$ cp oozie/oozie-client-4.3.0.tar.gz .
$ tar -xvf oozie-client-4.3.0.tar.gz
$ mv oozie-client-3.3.2 oozie-client
$ cd bin
```

Agregue $ HOME / oozie-client / bin a la variable PATH en el archivo .bashrc y reinicie su terminal o haga

```
source $HOME/.bashrc
```

Para obtener más detalles sobre la configuración, puede consultar esta URL https://oozie.apache.org/docs/4.3.0/DG_QuickStart.html

Ahora puede enviar trabajos de hadoop a oozie en su terminal.

Para ejecutar un ejemplo, puede seguir esta URL y configurar su primer ejemplo para ejecutar https://oozie.apache.org/docs/4.3.0/DG_Examples.html

Puede enfrentar el error de abajo mientras ejecuta el ejemplo de reducción de mapa en la URL anterior

> java.io.IOException: java.net.ConnectException: la llamada desde localhost.localdomain / 127.0.0.1 a 0.0.0.0:10020 falló en la excepción de conexión: java.net.ConnectException: conexión rechazada; Para más detalles vea: http://wiki.apache.org/hadoop/ConnectionRefused

Solución: Inicie mr-jobhistory-server.sh

```
cd $HADOOP_HOME/sbin
./mr-jobhistory-server.sh start historyserver
```

Otro punto a tener en cuenta sobre la modificación del archivo job.properties es:

```
nameNode=hdfs://localhost:9000
jobTracker=localhost:8032
```

en su caso, esto puede ser diferente, ya que estoy usando apache hadoop, puede estar usando cloudera / hdp / cualquiera

```
To run spark job, I have tried running in local[*], yarn-client and
yarn-cluster as master, but succeeded in local[*] only
```

Lea Empezando con oozie en línea: https://riptutorial.com/es/oozie/topic/3437/empezando-con-

oozie

Capítulo 2: Coordinador de datos activados por Oozie

Introducción

Se proporciona una explicación detallada sobre el trabajo del coordinador activado con los datos de Oozie con un ejemplo.

El coordinador se ejecuta periódicamente desde la hora de inicio hasta la hora de finalización. A partir de la hora de inicio, el coordinador verifica si los datos de entrada están disponibles. Cuando los datos de entrada están disponibles, se inicia un flujo de trabajo para procesar los datos de entrada que, al finalizar, producen los datos de salida requeridos. Este proceso se repite en cada tic de frecuencia hasta la hora final del coordinador.

Observaciones

```
    <done-flag>_SUCCESS</done_flag>
```

El fragmento de código anterior en coordinator.xml para el conjunto de datos de entrada señala la presencia de datos de entrada. Eso significa que la acción del coordinador estará en estado de ESPERA hasta que el archivo _SUCCESS esté presente en el directorio de entrada dado. Una vez que esté presente, el flujo de trabajo comenzará a ejecutarse.

Examples

oozie coordinator sample

El siguiente trabajo de coordinador activará la acción del coordinador una vez en un día que ejecuta un flujo de trabajo. El flujo de trabajo tiene un script de shell que mueve la entrada a la salida.

```
<coordinator-app name="log_process_coordinator" frequency="${coord:days(1)}" start="2017-04-
29T06:00Z" end="2018-04-29T23:25Z" timezone="UTC"
             xmlns="uri:oozie:coordinator:0.2">
<datasets>
    <dataset name="input_dataset" frequency="${coord:days(1)}" initial-instance="2017-04-
29T06:00Z" timezone="GMT">
        <uri-template>${nameNode}/mypath/coord_job_example/input/${YEAR}${MONTH}${DAY}</uri-
template>
        <done-flag>_SUCCESS</done-flag>
    </dataset>
    <dataset name="output_dataset" frequency="${coord:days(1)}" initial-instance="2017-04-
29T06:00Z" timezone="GMT">
        <uri-template>${nameNode}/mypath/coord_job_example/output/${YEAR}${MONTH}${DAY}</uri-
template>
        <done-flag>_SUCCESS</done-flag>
```

```
    </dataset>
</datasets>
<input-events>
    <data-in name="input_event" dataset="input_dataset">
        <instance>${coord:current(0)}</instance>
    </data-in>
</input-events>
 <output-events>
    <data-out name="output_event" dataset="output_dataset">
        <instance>${coord:current(0)}</instance>
    </data-out>
</output-events>
 <action>
    <workflow>
        <app-path>${workflowAppUri}</app-path>
        <configuration>
            <property>
                <name>jobTracker</name>
                <value>${jobTracker}</value>
            </property>
            <property>
                <name>nameNode</name>
                <value>${nameNode}</value>
            </property>
            <property>
                <name>pool.name</name>
                <value>${poolName}</value>
            </property>
            <property>
                <name>inputDir</name>
                <value>${coord:dataIn('input_event')}</value>
            </property>
             <property>
                <name>outputDir</name>
                <value>${coord:dataOut('output_event')}</value>
            </property>
        </configuration>
    </workflow>
</action>
```

</coordinator-app>

muestra de flujo de trabajo oozie

```
<workflow-app xmlns="uri:oozie:workflow:0.4" name="shell-wf">
<start to="shell-node"/>
<action name="shell-node">
 <shell xmlns="uri:oozie:shell-action:0.2">
<job-tracker>${jobTracker}</job-tracker>
<name-node>${nameNode}</name-node>
<configuration>
 <property>
   <name>mapred.job.queue.name</name>
   <value>${poolName}</value>
 </property>
</configuration>
<exec>${myscript}</exec>
<argument>${inputDir}</argument>
<argument>${outputDir}</argument>
```

```
<file>${myscriptPath}</file>
<capture-output/>
</shell>
<ok to="end"/>
<error to="fail"/>
</action>
 <kill name="fail">
  <message>Shell action failed, error message[${wf:errorMessage(wf:lastErrorNode())}]
  </message>
</kill>
 <end name="end"/>
</workflow-app>
```

muestra de job.properties

```
nameNode=hdfs://namenode:port
start=2016-04-12T06:00Z
end=2017-02-26T23:25Z
jobTracker=yourjobtracker
poolName=yourpool
oozie.coord.application.path=${nameNode}/hdfs_path/coord_job_example/coord
workflowAppUri=${oozie.coord.application.path}
myscript=myscript.sh
myscriptPath=${oozie.coord.application.path}/myscript.sh
```

muestra de shell script

```
inputDir=${1}
outputDir=${2}
hadoop fs -mkdir -p ${outputDir}
hadoop fs -cp ${inputDir}/* ${outputDir}/
```

presentando el trabajo de coordinador

Copie el script, coordinator.xml y workflow.xml en HDFS. coordinator.xml debe estar presente en el directorio especificado por oozie.coord.application.path en job.properties. workflow.xml debe estar presente en el directorio especificado por workflowAppUri. Una vez que todo esté en su lugar, ejecute el siguiente comando desde el shell

```
oozie job -oozie <oozie_url>/oozie/ -config job.properties
```

Lea Coordinador de datos activados por Oozie en línea:
https://riptutorial.com/es/oozie/topic/9845/coordinador-de-datos-activados-por-oozie

Capítulo 3: Oozie 101

Examples

Arquitectura Oozie

Oozie está desarrollado en una arquitectura cliente-servidor. El servidor Oozie es una aplicación web de Java que ejecuta el contenedor de servlet de Java dentro de un Apache Tomcat integrado. Oozie proporciona tres tipos diferentes de clientes para interactuar con el servidor de Oozie: línea de comandos, API de cliente de Java y API de REST de HTTP.

El servidor Oozie no almacena ninguna información en memoria de los trabajos en ejecución. Se basa en RDBMS para almacenar estados y datos de todos los trabajos de Oozie. Cada vez que recupera la información del trabajo de la base de datos y almacena la información actualizada en la base de datos.

Oozie Server (can) se encuentra fuera del clúster de Hadoop y realiza la orquestación de los trabajos de Hadoop definidos en un trabajo de flujo de trabajo de Oozie.

Implementación de la aplicación Oozie

La aplicación Oozie más sencilla consiste en un archivo lógico de flujo de trabajo (workflow.xml), un archivo de propiedades de flujo de trabajo (job.properties/job.xml) y archivos JAR, scripts y archivos de configuración necesarios. Excepto el archivo de propiedades del flujo de trabajo, todos los demás archivos deben almacenarse en una ubicación HDFS. El archivo de propiedades del flujo de trabajo debe estar disponible localmente, desde donde se envía y se inicia la aplicación Oozie.

El directorio HDFS, donde se almacena workflow.xml junto con otros scripts y archivos de configuración, se denomina directorio de aplicaciones de flujo de trabajo Oozie. Todos los archivos JAR deben almacenarse en un directorio / lib en el directorio de la aplicación oozie.

Las aplicaciones Oozie más complejas pueden consistir en coordinadores (coordinator.xml) y archivos de lógica de paquete (bundle.xml). Estos archivos también se almacenan en el HDFS en un directorio de aplicación Oozie respectivo.

Cómo pasar la configuración con el envío de trabajos de Oozie Proxy

Cuando se utiliza la API de envío de trabajos de Oozie Proxy para enviar las acciones Oozie `Hive` , `Pig` y `Sqoop` . Para pasar cualquier configuración a la acción, se requiere que esté en el siguiente formato.

Para la acción de Hive:

- oozie.hive.options.size: la cantidad de opciones que pasará a la acción de Hive.
- oozie.hive.options.n: un argumento para pasar a Hive, la 'n' debe ser un número entero que

comience con cero (0) para indicar el número de opción.

```
<property>
    <name>oozie.hive.options.1</name>
    <value>-Doozie.launcher.mapreduce.job.queuename=hive</value>
</property>
<property>
    <name>oozie.hive.options.0</name>
    <value>-Dmapreduce.job.queuename=hive</value>
</property>
<property>
    <name>oozie.hive.options.size</name>
    <value>2</value>
</property>
```

Para la acción del cerdo:

- oozie.pig.options.size: la cantidad de opciones que pasará a la acción de Pig.
- oozie.pig.options.n: un argumento para pasar a Pig, la 'n' debe ser un entero que comience con cero (0) para indicar el número de opción.

```
<property>
    <name>oozie.pig.options.1</name>
    <value>-Doozie.launcher.mapreduce.job.queuename=pig</value>
</property>
<property>
    <name>oozie.pig.options.0</name>
    <value>-Dmapreduce.job.queuename=pig</value>
</property>
<property>
    <name>oozie.pig.options.size</name>
    <value>2</value>
</property>
```

Para la acción de Sqoop:

- oozie.sqoop.options.size: la cantidad de opciones que pasará al trabajo de Sqoop Hadoop.
- oozie.sqoop.options.n: Un argumento para pasar a Sqoop. hadoop job conf, la 'n' debe ser un número entero que comience con cero (0) para indicar el número de opción.

```
<property>
    <name>oozie.sqoop.options.1</name>
    <value>-Doozie.launcher.mapreduce.job.queuename=sqoop</value>
</property>
<property>
    <name>oozie.sqoop.options.0</name>
    <value>-Dmapreduce.job.queuename=sqoop</value>
</property>
<property>
    <name>oozie.sqoop.options.size</name>
    <value>2</value>
</property>
```

Lea Oozie 101 en línea: https://riptutorial.com/es/oozie/topic/4134/oozie-101

Creditos

S. No	Capítulos	Contributors
1	Empezando con oozie	Community, Jyoti Ranjan, YoungHobbit
2	Coordinador de datos activados por Oozie	sunitha
3	Oozie 101	YoungHobbit