

# Automatización de Jobs Hadoop con Oozie

Cristian Muñoz R,

Big Data Week Santiago, Chile

# Agenda

Sobre Cristian

Que es Apache Oozie

Arquitectura

Conceptos principales

Workflow Actions

Ejemplos



# Sobre Cristian

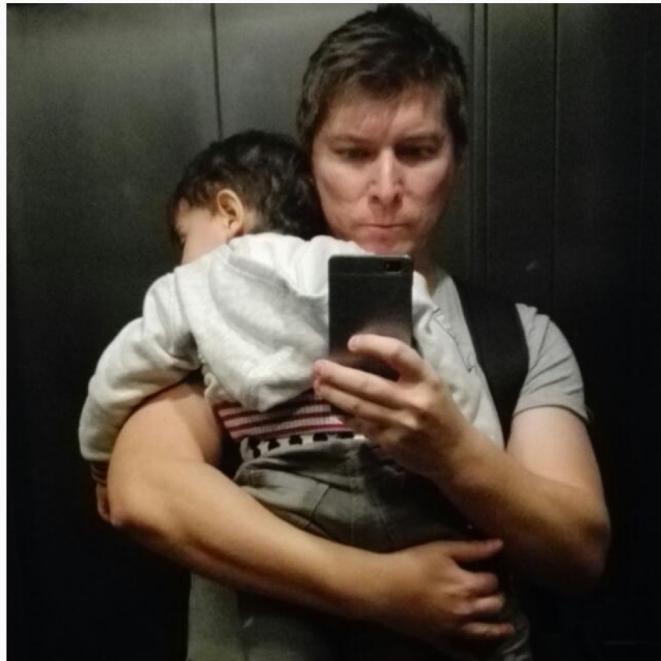
Ingeniería Informática , INACAP

7 años de experiencia en BI

4 años de experiencia en Big Data

Participación en proyectos Big Data  
área financiera, salud y retail

Experiencia en HortonWorks &  
Cloudera



### El comienzo

- El creador fue Yahoo
- Nace de la necesidad de organizar gran cantidad de jobs.
- Todo se complicó con la necesidad de dependencias entre procesos.
- Difícil detección de los errores.
- En el 2008 se crea la primera version de Oozie.
- En el 2010 pasa a ser un proyecto Apache.

# El comienzo

## *Major releases*

- 1.x Soporte para workflows (2010)(1.6.2)
- 2.x Soporte para coordinators (2011)(2.3.2)
- 3.x Soporte para bundles (2013)(3.3.2)
- 4.x Soporte para SSH, Hive, Sqoop, DistCp, Java, Shell, email.  
Diferentes bases de datos Derby, MySQL, PostgreSQL, Oracle (2014)  
(4.1.0).
- 4.3.0 (2016)
- ?

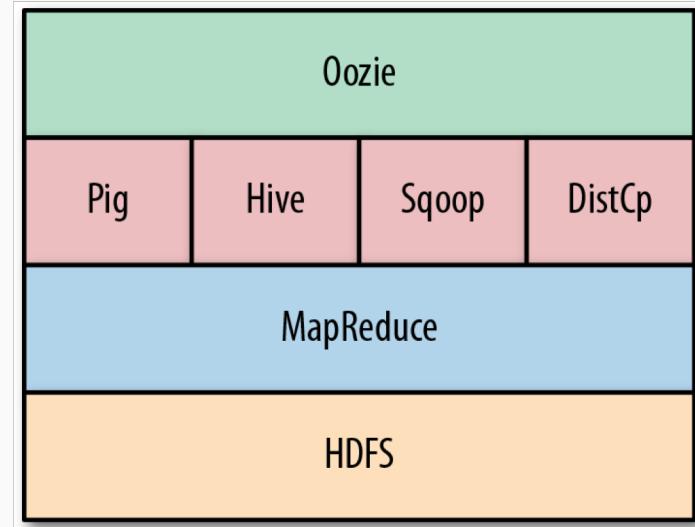
## Que es Apache Oozie

- Es un sistema de orquestación para jobs hadoop.
- Pueden correr bajo demanda o periódicamente
- Una aplicación oozie es multistage
- Un oozie job que corre bajo demanda se llama *workflow*.
- Un oozie job que corre periodicamente es llamado *coordinator*.
- Un conjunto de workflows es llamado coordinator (\*)
- Un conjunto de coordinators es llamado *bundle*

# Que es Apache Oozie

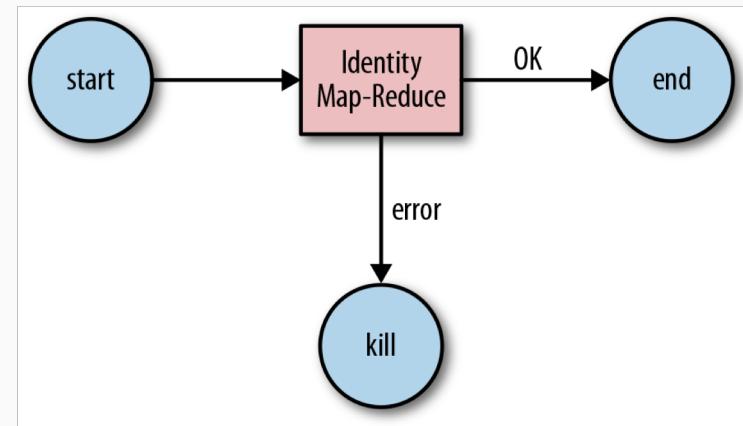
## Que hace

- Permite la ejecución de jobs map reduce
- Permite la ejecución de comandos para realizar operaciones en HDFS
- Permite ejecutar jobs que no son Map Reduce como por ejemplo la ejecución de un shell script.



## Un job simple (workflow)

- Tiene un inicio llamado start.
- Ejecuta un job map reduce.
- La aplicación oozie puede terminar correctamente (end)  
o con errores (kill)
- Ver ejemplo [\*\*primero\\_workflow.xml\*\*](#)



## Como se ejecuta

- Estructura en HDFS

```
app/
  |
  |-- workflow.xml
```

```
$ hdfs dfs -put target/example/ch01-identity ch01-identity
$ hdfs dfs -ls -R ch01-identity
```

- Como quedó la estructura en HDFS

```
/user/joe/ch01-identity/app
/user/joe/ch01-identity/app/workflow.xml
/user/joe/ch01-identity/data
/user/joe/ch01-identity/data/input
/user/joe/ch01-identity/data/input/input.txt
```

- Archivo properties (FS)

```
nameNode=hdfs://localhost:8020
jobTracker=localhost:8032
exampleDir=${nameNode}/user/${user.name}/ch01-identity
oozie.wf.application.path=${exampleDir}/app
```

- Ejecución de job

```
$ export OOZIE_URL=http://localhost:11000/oozie
$ oozie job -run -config target/example/job.properties
job: 0000006-130606115200591-oozie-joe-W
```

## Resultados

Solimos status del job oozie

```
# oozie job -info 0000006-130606115200591-oozie-joe-W
```

```
-----  
Workflow Name : identity-WF  
App Path     : hdfs://localhost:8020/user/joe/ch01-identity/app  
Status       : SUCCEEDED  
Run          : 0  
User         : joe  
Group        : -  
Created       : 2013-06-06 20:35 GMT  
Started       : 2013-06-06 20:35 GMT  
Last Modified : 2013-06-06 20:35 GMT  
Ended         : 2013-06-06 20:35 GMT  
CoordAction ID: -
```

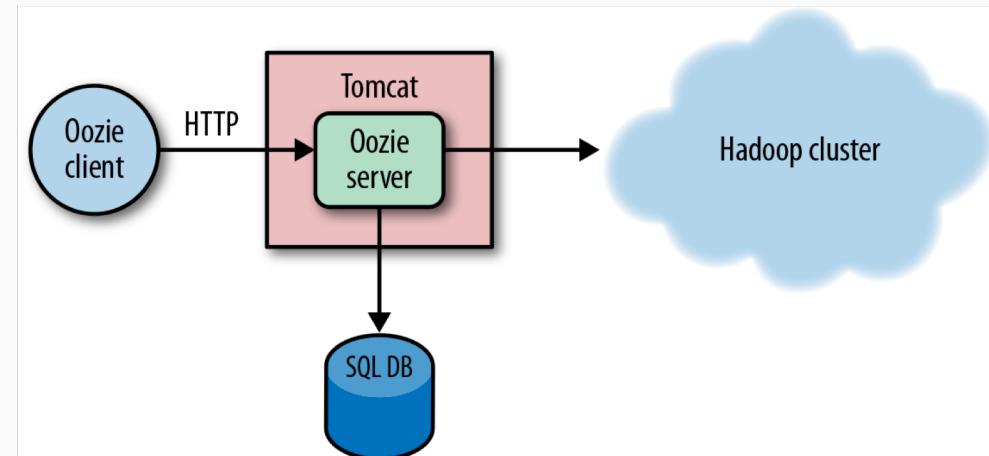
### Actions

ID	Status
0000006-130606115200591-oozie-joe-W@start:	OK
0000006-130606115200591-oozie-joe-W@identity-MR	OK
0000006-130606115200591-oozie-joe-W@success	OK

# Arquitectura

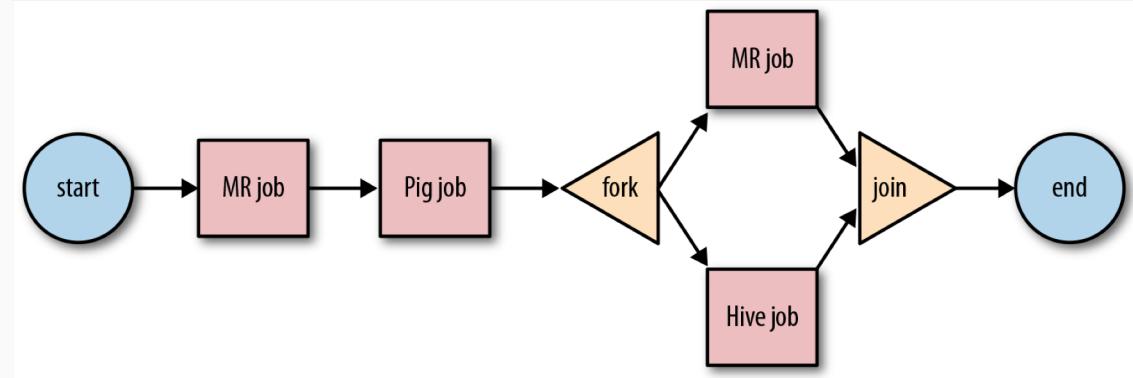
## Arquitectura

- API Java y la linea de comandos de Oozie utilizan HTTP REST API para la comunicación.
- Toda la información como usuarios o información de jobs es almacenada en una base de datos. Nada se almacena en memoria.
- Cada consulta respecto al estado de un job es sobre la base de datos. (Derby, MySQL, Oracle y PostgreSQL)



## Workflows

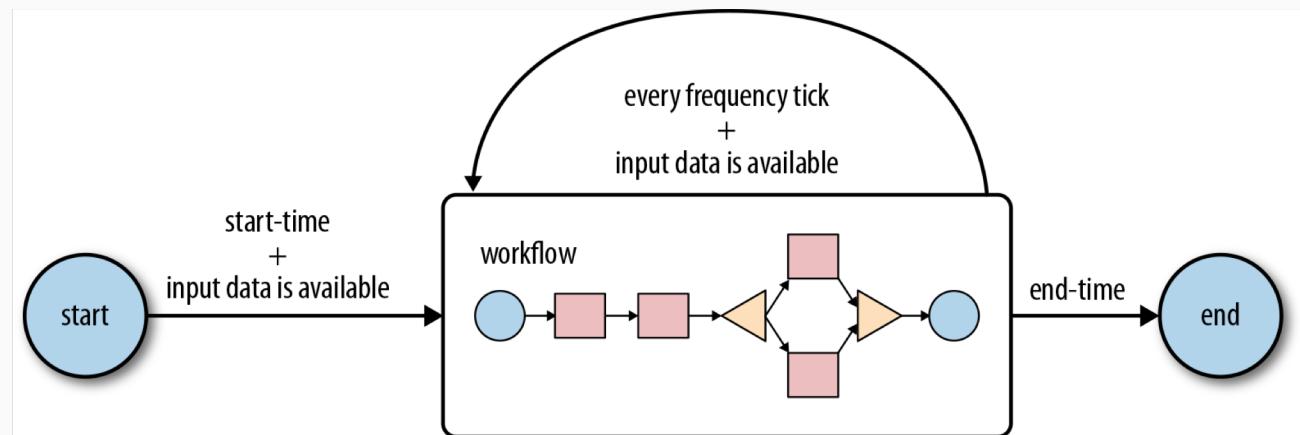
- Colección de acciones ordenadas en un DAG (directed acyclic graph)
- Cada una de las acciones generalmente es un job Hadoop, pero también puede existir acciones no del tipo Hadoop jobs como shell scripts, aplicaciones java o notificaciones de email).



# Conceptos principales

## Coordinators

- Calendariza la ejecución de workflows basado en fecha-hora de inicio y parámetro de frecuencia.
- Puede ejecutar un workflows en disponibilidad de data.



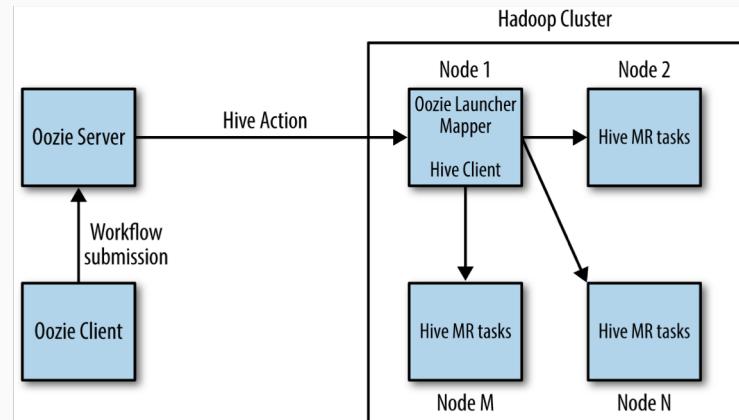
# Workflow Actions

## Que son

- Un conjunto de actions forman un job oozie.
- Existen acciones Hadoop y otras que realizan tareas con propósitos generales.

## Modelo de Ejecución

- El job oozie crea un job MapReduce el cual es ejecutado en el cluster a través de launcher job. (Cliente Oozie)
- El job MapReduce esta compuesto por 1 mapper.
- El Cliente Oozie realiza un submit al Oozie Server el cual podria o no estar en la misma máquina
- Finalmente Oozie server invoca el job basado en la libreria elegida. (hive, pig, sqoop, etc)
- Oozie nunca ejecuta ningun tipo de job en el mismo servidor. Con eso evita cualquier sobrecarga del mismo y entrega esa responsabilidad al cluster hadoop.



## Listado de actions soportados

- MapReduce action
- Java action
- Pig action
- FS action
- Sub-Workflow action
- Hive action
- Distcp action
- Email action
- SSH action
- Sqoop action

Nota: Spark action fue agregado en la version 4.2 de Apache Oozie.

### Ejemplo action MapReduce



```
# hadoop jar /user/joe/myApp.jar myAppClass \
-Dmapred.job.reduce.memory.mb=8192 \
/hdfs/user/joe/input \
/hdfs/user/joe/output
```

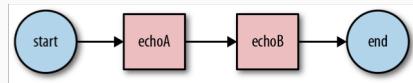
## Ejemplo action Sqoop

```
# sqoop import --connect jdbc:mysql://mysqlhost.mycompany.com/MY_DB \
--table test_table \
-username mytestsqoop \
-password password \
--target-dir /hdfs/joe/sqoop/output-data -m 1
```

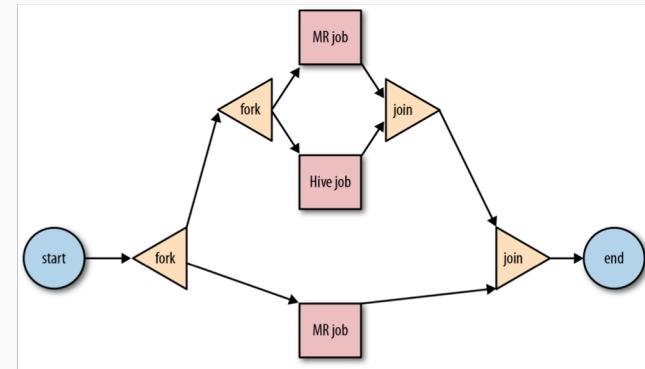
# Workflow Actions

## Nodos de control

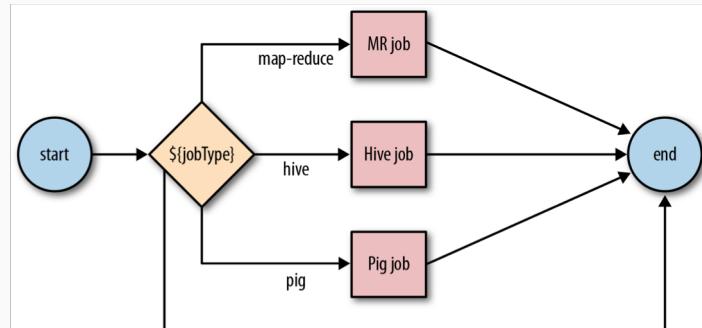
- <start> and <end>



- <fork> and <join> . Cuando no depende un action del otro. Aumenta velocidad de ejecución.



- <decision>. A diferencia de fork, solo uno se ejecuta. Cuando queremos aplicar if then else.

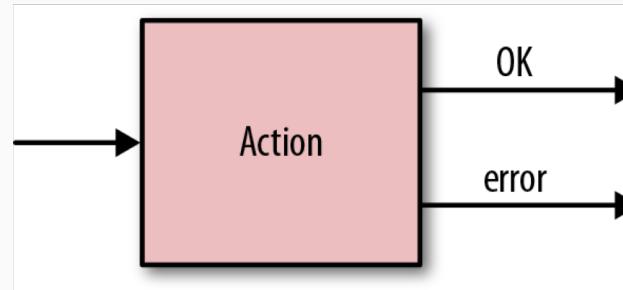


# Workflow Actions

## Nodos de control

- <kill>. Termina la ejecución del Job Oozie.
- <OK> and <ERROR>

```
<workflow-app xmlns="uri:oozie:workflow:0.4" name="killNodeWF">
<start to="mapReduce"/>
<action name="mapReduce">
  ...
  <ok to="done"/>
  <error to="error"/>
</action>
<kill name="error">
  <message>The 'mapReduce' action failed!</message>
<end name="done"/>
</workflow-app>
```



## Variables

- Parametrización de valores dentro de un Oozie job.

```
<job-tracker>${jobTracker}</job-tracker>
<name-node>${nameNode}</name-node>
```

## Funciones

- Cuando tenemos la necesidad de obtener valores que cambian dinámicamente en el transcurso de la ejecución.

### Ejemplos

- String wf:id(). Retorna el nombre del workflow.
- String timestamp(). Hora actual
- boolean fs:fileSize(String path). Tamaño en bytes de un path en HDFS.

## Archivo job properties

- Archivo el cual es invocado al momento de realizar un submit de un job oozie.

```
# oozie job -oozie http://localhost:4080/oozie/ -config ~/job.properties –run
```

- Siempre debe estar presente en FS. Nunca en HDFS.
- Corresponde a un set de argumentos que se traspasan a un job oozie.
- Ejemplo job.properties

```
nameNode= hdfs://localhost:8020
jobTracker=localhost:8032
queueName=research
oozie.use.system.libpath=true
oozie.wf.application.path=${nameNode}/user/joe/oozie/mrJob/firstWorkflow.xml
```

- Ejemplo de opción por línea de comandos

```
# oozie job -oozie http://localhost:4080/oozie/ -DqueueName=research -config job.properties –run
```

# Coordinator

## Características

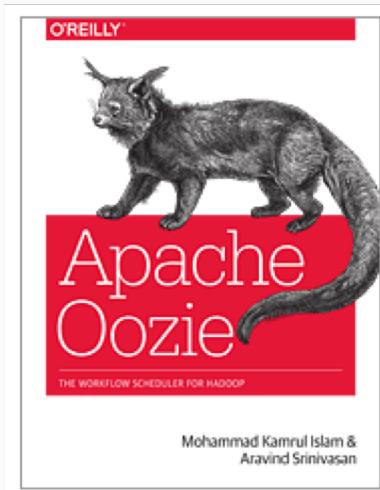
- Un job coordinator consta de un coordinator.xml y su job.properties
- Permiten ejecutar workflows basado en distintas formas de ejecución. Recurrentes en el tiempo o por dependencia de data.
- Los parametros definidos en un job.properties para un coordinator serán heredados por un workflow.
- El mismo coordinator puede ser ejecutado varias veces con diferentes valores. Por ejemplo hora de inicio y frecuencia de ejecución.
- Ejemplo coordinator.properties

```
nameNode=hdfs://localhost:8020  
jobTracker=localhost:8032  
appBaseDir=${nameNode}/user/${user.name}/ch06-first-coord  
oozie.coord.application.path=${appBaseDir}/app
```

# Conclusión

Conclusiones finales

## Donde la obtengo



<https://oozie.apache.org/docs/4.3.0/index.html>

# Gracias!

<https://www.linkedin.com/in/cgmuros/>

Big Data Week Chile

