

Oozie is an orchestration system for Hadoop Jobs.

Oozie is a Java web application used for Scheduling hadoop MR jobs.

Workflow means Group of tasks which has to be executed in a kind of order, it can be sequential or it can be parallel.

Directed Acyclic Graph, Acyclic because it doesn’t have any cycles

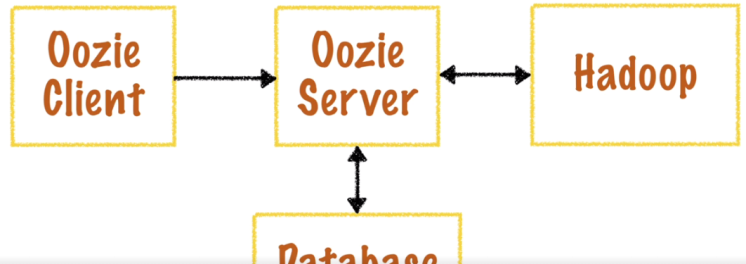
Coordinator schedules the execution of the workflow at a specific time and/or specified frequency.

* If the Input data is not available then the workflow will be delayed till the data becomes available.
* If no input data is required then the workflow runs purely at a specified time or frequency

Output of one Coordinator job managing a workflow can be the input to another Coordinator job, Thus we are forming a data Pipelines. Data Pipeline involves transforming data in phases till the final output occurs.

Oozie is a single XML file which describes the entire Application; an Application can take any Forms

Workflow.xml, coordinator.xml, bundle.xml files describes their corresponding applications. These Files along with the required jar or files for the Oozie applications are copied to the HDFS before the job run.



Oozie Server

* Manages all the Oozie job scheduling and job execution
* Its wont store anything , it stores all the workflow coordinator bundle related information in the database

Oozie Client

* It’s what we use. Typically it would be a command line
* Oozie client can be written in any languages

Hadoop

* Oozie Application read the xml from the HDFS
* It run the jobs on the Hadoop Cluster

Database

* All job related information stored in the database
* Oozie server queries this database in order to retrieve any info about the job.
* It may be derby(default) mysql oracle

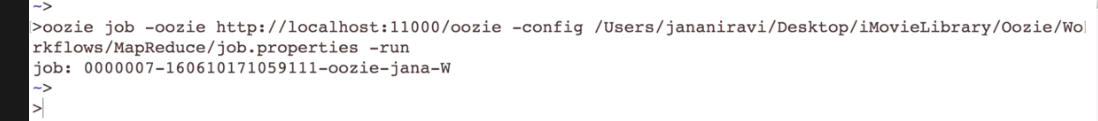
Installation

Running First Application

Work Flow is the Fundamental Building block of Oozie.

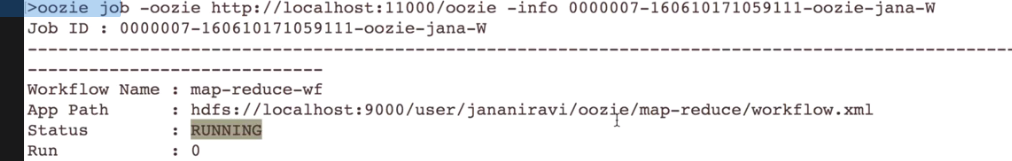
Workflow is responsible for bringing different action together. Actions are Hadoop Jobs.

>oozie job –oozie [http://localhost:11000/oozie -config /oozie/job.properties](http://localhost:11000/oozie%20-config%20/oozie/job.properties) –run



Export OOZIE\_URL=http://localhost:11000/oozie if we set this we can remove –oozie flag

Checking the Job



Job.properties

It contains the job Configuration to send to Oozie to invoke the workflow.

It contains the Arguments for the Oozie workflow application.

nameNode=hdfs://localhost:9000 Specify the Name Node

jobTracker=localhost:8032 in Hadoop 2.o we have to specify the Resource Manager URL

In order to make Oozie as a flexible we are Specify some Parameters which we can used in the upcoming Workflow XML File

queueName=default

oozieRoot=oozie

oozie.system.libpath=true

Oozie.system.libpath=true this will heck for the jars files at the sharelib path , For running Hive jobs this flag must be specified

oozie.wf.application.path=${nameNode}/user/${user.name}/${oozieRoot}/map-reduce/workflow.xml

Specifying the Work Flow xml Path

WorkFlow.xml

As of now we see simple workflow, but in reality we have many actions within the workflow. Then we have to mention whether the Action needs to run in parallel or serial.

1. Control Nodes
   1. These control the start and end and basic execution of the workflow

Complex Examples are Fork and Join

By using Fork we can specify which actions to be run in parallel. By using Joins we can specify when this parallel job finishes and Continues to the next serial job

Decision Node

Based on the Decision, one action has to be performed out of the 3 actions

<decision name= “decision”>

<switch >

<case to=”MR”>

${jobtype eq “MR”}

</case>

<default to=”ok”>

</switch>

1. Action Nodes

Action nodes specify the actual processing needs to done by the workflow it could be a <map-reduce> <shell> <python> <hive> <email>

Actions are Asynchronous except the FS

Actions which are completed transferred to the <ok> node or to the <error> node

1. Global Configuration

Many Parameters may be common across all the Actions in the Workflow. We can bring the same under the <global> node

Lets Run the Work Flow based on the Time and the data triggers

Coordinators - we are having two Triggers

Time trigger

Data Trigger

Materialize a workflow means submitting the workflow and executing it

Coordinator supports only one work Flow.

Examples :1

Time Based Trigger :- A unix cron Job

Coordinator Action is the Workflow this coordinator manages

In order to avoid overload the Hadoop Cluster, we have some Control Mechanics for the Co-coordinator

1. Throttle – maximum no of jobs that can be in the waiting state at any time by default 12
2. Timeout – maximum time a coordinator action can be in the waiting state by default 7 days
3. Execution order – which one is chosen first FIFO (WORKING GOOD)LIFO LAST\_ONLY
4. Concurrency – How many Actions/jobs can run on Hadoop simultaneously by default is 1

