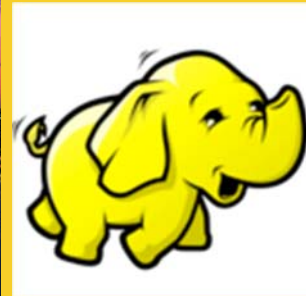




# Oozie

Originals of slides and source code for examples: <http://www.coreservlets.com/hadoop-tutorial/>  
Also see the customized Hadoop training courses (onsite or at public venues) – <http://courses.coreservlets.com/hadoop-training.html>

**Customized Java EE Training: <http://courses.coreservlets.com/>**  
Hadoop, Java, JSF 2, PrimeFaces, Servlets, JSP, Ajax, jQuery, Spring, Hibernate, RESTful Web Services, Android.  
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.



**For live customized Hadoop training (including prep for the Cloudera certification exam), please email [info@coreservlets.com](mailto:info@coreservlets.com)**

**Taught by recognized Hadoop expert who spoke on Hadoop several times at JavaOne, and who uses Hadoop daily in real-world apps. Available at public venues, or customized versions can be held on-site at your organization.**

- Courses developed and taught by Marty Hall
  - JSF 2.2, PrimeFaces, servlets/JSP, Ajax, jQuery, Android development, Java 7 or 8 programming, custom mix of topics
  - Courses available in any state or country. Maryland/DC area companies can also choose afternoon/evening courses.
- Courses developed and taught by [coreservlets.com](http://coreservlets.com) experts (edited by Marty)
  - Spring, Hibernate/JPA, GWT, Hadoop, HTML5, RESTful Web Services

Contact [info@coreservlets.com](mailto:info@coreservlets.com) for details



# Agenda

- **Introduce Oozie**
- **Oozie Installation**
- **Write Oozie Workflow**
- **Deploy and Run Oozie Workflow**

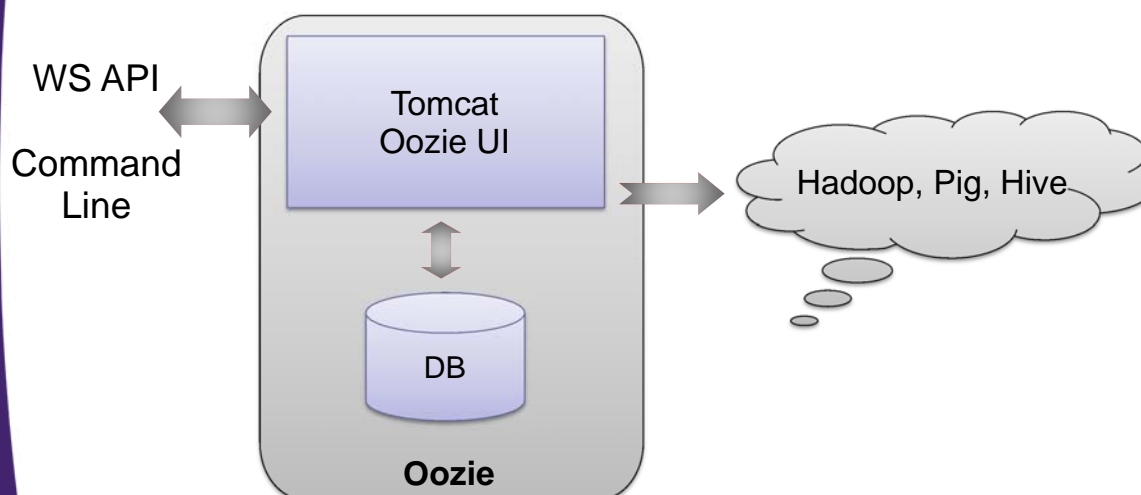
4

# Oozie

- **Workflow scheduler for Hadoop**
  - Java MapReduce Jobs
  - Streaming Jobs
  - Pig
- **Top level Apache project**
  - Comes packaged in major Hadoop Distributions
    - Cloudera Distribution for Hadoop (CDH)
  - <http://incubator.apache.org/oozie>
- **Provides workflow management and coordination of those workflows**
- **Manages Directed Acyclic Graph (DAG) of actions**

5

# Oozie



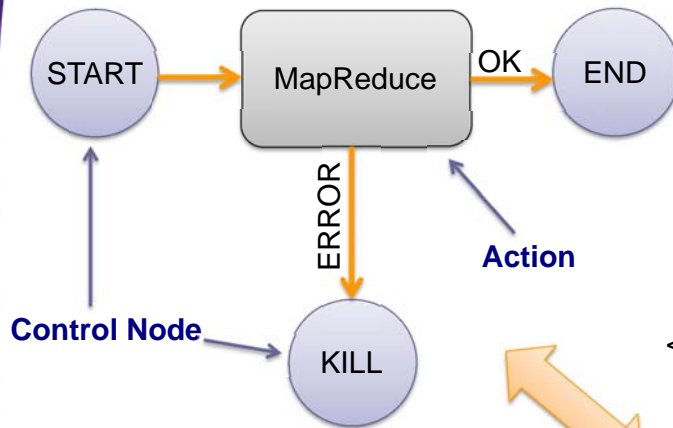
6

# Oozie

- **Runs HTTP service**
  - Clients interact with the service by submitting workflows
  - Workflows are executed immediately or later
- **Workflows are defined via XML**
  - Instead of writing Java code that implements Tool interface and extending Configured class

7

# Action and Control Nodes



```
<workflow-app name="foo-wf"..  
  <start to="[NODE-NAME]"/>  
  <map-reduce>  
    ...  
  </map-reduce>  
  <kill name="[NODE-NAME]">  
    <message>Error occurred</message>  
  </kill>  
  <end name="[NODE-NAME]"/>  
</workflow-app>
```

8

# Action and Control Nodes

- **Control Flow**

- start, end, kill
- decision
- fork, join

- **Actions**

- map-reduce
- java
- pig
- hdfs

9

# Oozie Coordination Engine

- **Oozie Coordination Engine can trigger workflows by**
  - Time (Periodically)
  - Data Availability (Data appears in a directory)

10

# Install Oozie

- **\$ mkdir <OOZIE\_HOME>/libext**
- **Download ExtJS and place under <OOZIE\_HOME>/libext**
  - ext-2.2.zip
- **Place Hadoop libs under libext**
  - \$ cd <OOZIE\_HOME>
  - \$ tar xvf oozie-hadooplibs-3.1.3-cdh4.0.0.tar.gz
  - \$ cp oozie-3.1.3-cdh4.0.0/hadooplibs/hadooplib-2.0.0-cdh4.0.0/\*.jar libext/
- **Configure Oozie with components under libext**
  - \$ bin/oozie-setup.sh

11

## Install Oozie

- **Create environment variable for default url**

- export OOOZIE\_URL=<http://localhost:11000/oozie>
- This allows you to use \$oozie command without providing url

- **Update oozie-site.xml to point to Hadoop configuration**

```
<property>
  <name>oozie.service.HadoopAccessorService.hadoop.configurations</name>
  <value>*=/home/hadoop/Training/CDH4/hadoop-2.0.0-cdh4.0.0/conf</value>
</property>
```

- **Setup Oozie database**

- \$./bin/ooziedb.sh create -sqlfile oozie.sql -run DB Connection.

12

## Install Oozie

- **Update core-site.xml to allow Oozie become “hadoop” and for that user to connect from any host**

```
<property>
  <name>hadoop.proxyuser.hadoop.groups</name>
  <value>*</value>
  <description>Allow the superuser oozie to impersonate any members of the group group1 and group2</description>
</property>
<property>
  <name>hadoop.proxyuser.hadoop.hosts</name>
  <value>*</value>
  <description>The superuser can connect only from host1 and host2 to impersonate a user</description>
</property>
```

- **Learn more:**

[http://hadoop.apache.org/common/docs/r1.0.3/Secure\\_Impersonation.html](http://hadoop.apache.org/common/docs/r1.0.3/Secure_Impersonation.html)

13



# Start Oozie

```
$ oozie-start.sh
```

```
Setting OOZIE_HOME:      /home/hadoop/Training/CDH4/oozie-3.1.3-cdh4.0.0
Setting OOZIE_CONFIG:    /home/hadoop/Training/CDH4/oozie-3.1.3-cdh4.0.0/conf
Sourcing:                /home/hadoop/Training/CDH4/oozie-3.1.3-cdh4.0.0/conf/oozie-env.sh
  setting OOZIE_LOG=/home/hadoop/Training/logs/oozie
  setting CATALINA_PID=/home/hadoop/Training/hadoop_work/pids/oozie.pid
Setting OOZIE_CONFIG_FILE: oozie-site.xml
Setting OOZIE_DATA:      /home/hadoop/Training/CDH4/oozie-3.1.3-cdh4.0.0/data
Using OOZIE_LOG:         /home/hadoop/Training/logs/oozie
Setting OOZIE_LOG4J_FILE: oozie-log4j.properties
Setting OOZIE_LOG4J_RELOAD: 10
Setting OOZIE_HTTP_HOSTNAME: localhost
Setting OOZIE_HTTP_PORT: 11000
Setting OOZIE_ADMIN_PORT: 11001
```

```
...
...
...
```

14

# Test Installation

```
$ oozie admin -status
System mode: NORMAL
```

<http://localhost:11000/oozie/>

Oozie Web Console - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://localhost:11000/oozie/

Most Visited Resources Hadoop Manage... Docs

Oozie Web Console

Documentation

Oozie Web Console

Workflow Jobs Coordinator Jobs Bundle Jobs System Info Instrumentation

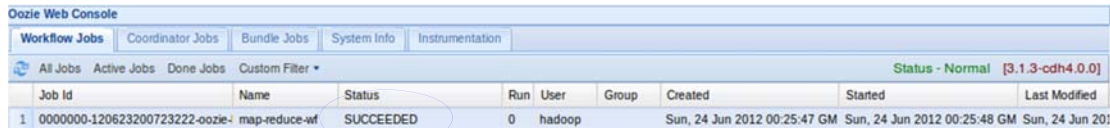
All Jobs Active Jobs Done Jobs Custom Filter Status - Normal [3.1.3-cdh4.0.0]

Job Id	Name	Status	Run	User	Group	Created	Started	Last Modified
1 0000026-120623200723222-oozie-most-seen-letter	most-seen-letter	SUCCEDEE	0	hadoop		Sun, 24 Jun 2012 04:31:58 GM	Sun, 24 Jun 2012 04:31:58 GM	Sun, 24 Jun 2012 04:32:51 C
2 0000025-120623200723222-oozie-most-seen-letter	most-seen-letter	SUCCEDEE	0	hadoop		Sun, 24 Jun 2012 04:23:32 GM	Sun, 24 Jun 2012 04:23:32 GM	Sun, 24 Jun 2012 04:24:27 C
3 0000024-120623200723222-oozie-most-seen-letter	most-seen-letter	KILLED	0	hadoop		Sun, 24 Jun 2012 04:20:32 GM	Sun, 24 Jun 2012 04:20:32 GM	Sun, 24 Jun 2012 04:20:45 C
4 0000023-120623200723222-oozie-most-seen-letter	most-seen-letter	KILLED	0	hadoop		Sun, 24 Jun 2012 04:17:42 GM	Sun, 24 Jun 2012 04:17:42 GM	Sun, 24 Jun 2012 04:17:55 C
5 0000022-120623200723222-oozie-most-seen-letter	most-seen-letter	SUCCEDEE	0	hadoop		Sun, 24 Jun 2012 04:12:05 GM	Sun, 24 Jun 2012 04:12:05 GM	Sun, 24 Jun 2012 04:13:01 C
6 0000021-120623200723222-oozie-most-seen-letter	most-seen-letter	KILLED	0	hadoop		Sun, 24 Jun 2012 04:04:29 GM	Sun, 24 Jun 2012 04:04:29 GM	Sun, 24 Jun 2012 04:05:44 C
7 0000020-120623200723222-oozie-most-seen-letter	most-seen-letter	SUCCEDEE	0	hadoop		Sun, 24 Jun 2012 03:17:13 GM	Sun, 24 Jun 2012 03:17:13 GM	Sun, 24 Jun 2012 03:17:44 C
8 0000019-120623200723222-oozie-most-seen-letter	most-seen-letter	KILLED	0	hadoop		Sun, 24 Jun 2012 03:12:14 GM	Sun, 24 Jun 2012 03:12:14 GM	Sun, 24 Jun 2012 03:12:55 C
9 0000018-120623200723222-oozie-most-seen-letter	most-seen-letter	KILLED	0	hadoop		Sun, 24 Jun 2012 03:09:11 GM	Sun, 24 Jun 2012 03:09:11 GM	Sun, 24 Jun 2012 03:09:49 C
10 0000017-120623200723222-oozie-most-seen-letter	most-seen-letter	SUCCEDEE	0	hadoop		Sun, 24 Jun 2012 03:05:51 GM	Sun, 24 Jun 2012 03:05:51 GM	Sun, 24 Jun 2012 03:06:22 C
11 0000016-120623200723222-oozie-most-seen-letter	most-seen-letter	KILLED	0	hadoop		Sun, 24 Jun 2012 03:02:14 GM	Sun, 24 Jun 2012 03:02:14 GM	Sun, 24 Jun 2012 03:02:53 C
12 0000015-120623200723222-oozie-most-seen-letter	most-seen-letter	KILLED	0	hadoop		Sun, 24 Jun 2012 02:58:26 GM	Sun, 24 Jun 2012 02:58:26 GM	Sun, 24 Jun 2012 02:59:15 C
13 0000014-120623200723222-oozie-most-seen-letter	most-seen-letter	KILLED	0	hadoop		Sun, 24 Jun 2012 02:56:39 GM	Sun, 24 Jun 2012 02:56:39 GM	Sun, 24 Jun 2012 02:57:18 C
14 0000013-120623200723222-oozie-most-seen-letter	most-seen-letter	KILLED	0	hadoop		Sun, 24 Jun 2012 02:52:49 GM	Sun, 24 Jun 2012 02:52:49 GM	Sun, 24 Jun 2012 02:53:27 C

15

# Running Oozie Examples

- **Extract examples packaged with Oozie**
  - \$ cd \$OOZIE\_HOME
  - \$ tar xvf oozie-examples.tar.gz
- **Copy examples to HDFS to user's home directory**
  - \$ hdfs dfs -put examples examples
- **Run an example**
  - \$ oozie job -config examples/apps/map-reduce/job.properties -run
- **Check Web Console**
  - <http://localhost:11000/oozie/>



The screenshot shows the Oozie Web Console interface. At the top, there are tabs for Workflow Jobs, Coordinator Jobs, Bundle Jobs, System Info, and Instrumentation. Below the tabs, there are filters for All Jobs, Active Jobs, Done Jobs, and a Custom Filter dropdown. The main area displays a table of jobs. The table has columns for Job Id, Name, Status, Run, User, Group, Created, Started, and Last Modified. A single job is listed with Job Id 1, Name 0000000-120623200723222-oozie-map-reduce-wf, Status SUCCEEDED, Run 0, User hadoop, Group, Created Sun, 24 Jun 2012 00:25:47 GM, Started Sun, 24 Jun 2012 00:25:48 GM, and Last Modified Sun, 24 Jun 2012. The status 'SUCCEEDED' is circled in blue.

Job Id	Name	Status	Run	User	Group	Created	Started	Last Modified
1	0000000-120623200723222-oozie-map-reduce-wf	SUCCEEDED	0	hadoop		Sun, 24 Jun 2012 00:25:47 GM	Sun, 24 Jun 2012 00:25:48 GM	Sun, 24 Jun 2012

16

# Oozie Workflows

- **Defined in XML**
- **Uses Process Definition Language**
  - <http://incubator.apache.org/oozie/docs/3.2.0-incubating/docs/WorkflowFunctionalSpec.html>

```
<workflow-app name="foo-wf" xmlns="uri:oozie:workflow:0.1">
...
  <decision name="mydecision">
    <switch>
      <case to="reconsolidatejob">
        ${fs:fileSize(secondjobOutputDir) gt 10 * GB}
      </case>
      <case to="reexpandjob">
        ${fs:fileSize(secondjobOutputDir) lt 100 * MB}
      </case>
      <case to="recomputejob">
        ${hadoop:counters('secondjob')[RECORDS][REDUCE_OUT] lt 1000000}
      </case>
      <default to="end"/>
    </switch>
  </decision>
...
</workflow-app>
```

17

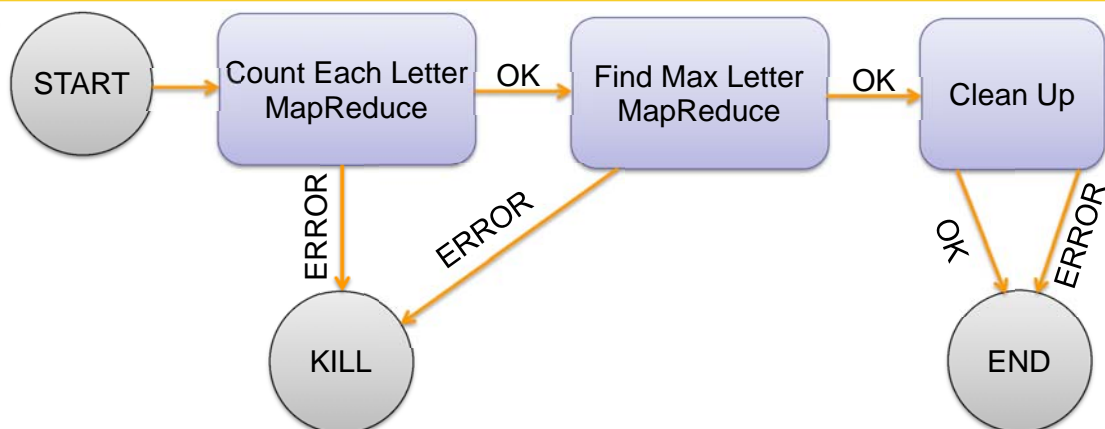


# Oozie Workflows

- **Workflows consist of**
  - Action nodes
    - MapReduce, Pig, Hive
    - Streaming, Java, etc...
  - Control flow nodes
    - Logic decisions between action nodes
    - Execute actions based on conditions or in parallel
- **Workflows begin with START node**
- **Workflows succeed with END node**
- **Workflows fail with KILL node**
- **Several actions support JSP Expression Language (EL)**

18

# Most Occurrences Workflows



Count Each Letter  
MapReduce

- **Action Node**



- **Control Flow Node**

END

- **Control Node**

This source is in *HadoopSamples*  
project under  
`/src/main/resources/mr/workflows`

19

# Most Occurrences Workflows

```
<workflow-app xmlns="uri:oozie:workflow:0.2" name="most-seen-letter">
  <start to="count-each-letter"/>
  <action name="count-each-letter">
    <map-reduce>
      <job-tracker>${jobTracker}</job-tracker>
      <name-node>${nameNode}</name-node>
      <prepare>
        <delete path="${nameNode}${outputDir}"/>
        <delete path="${nameNode}${intermediateDir}"/>
      </prepare>
      <configuration>
        ...
        <property>
          <name>mapreduce.job.map.class</name>
          <value>mr.wordcount.StartsWithCountMapper</value>
        </property>
        ...
      </configuration>
    </map-reduce>
    <ok to="find-max-letter"/>
    <error to="fail"/>
  </action>
  ...
</workflow-app>
```

START Action Node to count-each-letter MapReduce action

MapReduce have optional Prepare section

Pass property that will be set on MapReduce job's Configuration object

In case of success, go to the next job; in case of failure go to fail node

20

# First map-reduce Action

```
<map-reduce>
  <job-tracker>${jobTracker}</job-tracker>
  <name-node>${nameNode}</name-node>
  <prepare>
    <delete path="${nameNode}${outputDir}"/>
    <delete path="${nameNode}${intermediateDir}"/>
  </prepare>
  <configuration>
    <property>
      <name>mapred.mapper.new-api</name>
      <value>true</value>
    </property>
    <property>
      <name>mapred.reducer.new-api</name>
      <value>true</value>
    </property>
    <property>
      <name>mapred.job.queue.name</name>
      <value>${queueName}</value>
    </property>
    ...
  </configuration>
</map-reduce>
```

Administrative items to indicate where namenode and resource manager is

Optional prepare section; allows to execute command prior running the job

By default "old api" is used; specify to use new api

Specify which queue to submit this job to Resource Manager

21

# First map-reduce Action

```
...
<property>
  <name>mapreduce.job.map.class</name>
  <value>mr.wordcount.StartsWithCountMapper</value>
</property>
<property>
  <name>mapreduce.job.combine.class</name>
  <value>mr.wordcount.StartsWithCountReducer</value>
</property>
<property>
  <name>mapreduce.job.reduce.class</name>
  <value>mr.wordcount.StartsWithCountReducer</value>
</property>
<property>
  <name>mapreduce.job.inputformat.class</name>
  <value>org.apache.hadoop.mapreduce.lib.input.TextInputFormat</value>
</property>
<property>
  <name>mapreduce.job.outputformat.class</name>
  <value>org.apache.hadoop.mapreduce.lib.output.TextOutputFormat</value>
</property>
...
```

Specify Mapper, Reducer, Input and Output formats; this is instead of Tool implementation

This action will produce a file of tab separated key-value pairs as specified by TextOutputFormat

22

# First map-reduce Action (continued)

```
...
...
<property>
  <name>mapreduce.job.output.key.class</name>
  <value>org.apache.hadoop.io.Text</value>
</property>
<property>
  <name>mapreduce.job.output.value.class</name>
  <value>org.apache.hadoop.io.IntWritable</value>
</property>
<property>
  <name>mapreduce.input.fileinputformat.inputdir</name>
  <value>${inputFile}</value>
</property>
<property>
  <name>mapreduce.output.fileoutputformat.outputdir</name>
  <value>${intermediateDir}</value>
</property>
</configuration>
</map-reduce>
```

These properties are substituted from job.properties file

23

# Most Occurrences Workflows

```
<action name="find-max-letter"> ← Second MapReduce job
  <map-reduce>
    <job-tracker>${jobTracker}</job-tracker>
    <name-node>${nameNode}</name-node> ← Namenode and Yarn
    <configuration> ← Resource Manager Location
      ...
      ...
      <property> ← Token substituted from
        <name>mapreduce.job.map.class</name> application properties file
        <value>mr.workflows.MostSeenStartLetterMapper</value>
      </property>
      <property>
        <name>mapreduce.job.combine.class</name>
        <value>mr.workflows.MostSeendStartLetterReducer</value>
      ...
      ...
    </configuration>
  </map-reduce>
  <ok to="clean-up"/> ← Control Flow Node
  <error to="fail"/>
</action>
```

24

# Second map-reduce Action

```
...
<property>
  <name>mapreduce.job.map.class</name>
  <value>mr.workflows.MostSeenStartLetterMapper</value>
</property>
<property>
  <name>mapreduce.job.combine.class</name>
  <value>mr.workflows.MostSeendStartLetterReducer</value>
</property>
<property>
  <name>mapreduce.job.reduce.class</name>
  <value>mr.workflows.MostSeendStartLetterReducer</value>
</property>
...
```

↑  
Specify Mapper, Reducer and Combiner

25

## Second map-reduce Action (continued)

First map-reduce action produced a file with tab separated key-value pairs; second step utilizes `KeyValueTextInputFormat` to read these pairs as text

```
...
<property>
  <name>mapreduce.job.inputformat.class</name>
  <value>org.apache.hadoop.mapreduce.lib.input.KeyValueTextInputFormat</value>
</property>
<property>
  <name>mapreduce.job.outputformat.class</name>
  <value>org.apache.hadoop.mapreduce.lib.output.TextOutputFormat</value>
</property>
<property>
  <name>mapreduce.job.output.key.class</name>
  <value>org.apache.hadoop.io.Text</value>
</property>
<property>
  <name>mapreduce.job.output.value.class</name>
  <value>org.apache.hadoop.io.IntWritable</value>
</property>
...
```

26

## Most Occurrences Workflows

```
...
...
<action name="clean-up">
  <fs>
    <delete path='${nameNode}${intermediateDir}'/>
  </fs>
  <ok to="end"/>
  <error to="end"/>
</action>

<kill name="fail">
  <message>Map/Reduce failed, error
message[${wf:errorMessage(wf:lastErrorNode())}]</message>
</kill>

<end name="end"/>
</workflow-app>
```

Clean node, remove temporary folder

Workflow has failed, display error message

JSP expression language

Workflow ended with success

27



# Package and Run Your Workflow

- 1. Create application directory structure with workflow definitions and resources**
  - Workflow.xml, jars, etc..
- 2. Copy application directory to HDFS**
- 3. Create application configuration file**
  - specify location of the application directory on HDFS
  - specify location of the namenode and resource manager
- 4. Submit workflow to Oozie**
  - Utilize oozie command line
- 5. Monitor running workflow(s)**

28

## 1: Oozie Application Directory

- **Must comply to directory structure spec**

```
mostSeenLetter-oozieWorkflow
|--lib/
|   |--HadoopSamples.jar
|--workflow.xml
```

Application  
Workflow Root

Libraries should be placed  
under lib directory

Workflow.xml  
defines workflow

29

# 1: Oozie Application Directory

- **Can use a build tool to generate this structure**
  - Samples use maven plugins (see pom.xml)
    - Maven-dependency-plugin
    - Maven-resources-plugin
  - Run ‘mvn clean package’
    - Will create ‘mostSeenLetter-oozieWorkflow’ directory with dependencies and workflow definitions

30

# 2: Copy Application Directory to HDFS

- **Oozie utilizes HDFS to load applications**

```
hdfs dfs -put mostSeenLetter-oozieWorkflow
```

Copies directory from local files system onto HDFS;  
directory gets copied to user's home directory



31

## 3: Create Application Configuration File

- **job.properties** - Needs to exist locally, required for submission

```
nameNode=hdfs://localhost:8020
jobTracker=localhost:8021
queueName=default
```

Properties for required locations such as namenode and resource manage

```
inputFile=/training/data/hamlet.txt
intermediateDir=/training/playArea/mostSeenLetter-oozieWorkflow-tmp
outputDir=/training/playArea/oozieWorkflow
```

Properties needed for the MapReduce actions in the workflow

```
oozie.wf.application.path=${nameNode}/user/${user.name}/mostSeenLetter-oozieWorkflow
```

Most importantly HDFS location of the application must be specified

32

## 4: Submit Workflow to Oozie

- **Use oozie command line tool**

– For usage: \$oozie help

```
$ oozie job -config job.properties -run
job: 0000001-120711224224630-oozie-hado-W
```

Application configuration file

Application ID; use this ID to get status

33

# 5: Monitor Running Workflow(s)

- **Two options**

- Command line (\$oozie)
- Web Interface (<http://localhost:11000/oozie>)

34

# 5: Monitor Running Workflow(s) - Command Line

```
$ oozie job -info 0000001-120711224224630-oozie-hado-W  
Job ID : 0000001-120711224224630-oozie-hado-W
```

```
-----  
Workflow Name : most-seen-letter  
App Path      : hdfs://localhost:8020/user/hadoop/mostSeenLetter-oozieWorkflow  
Status       : RUNNING  
Run          : 0  
User         : hadoop  
Group        : -  
Created      : 2012-07-13 03:08  
Started      : 2012-07-13 03:08  
Last Modified : 2012-07-13 03:08  
Ended        : -  
CoordAction ID: -
```

Actions

```
-----  
ID                               Status  Ext ID          Ext Status Err Code  
-----  
0000001-120711224224630-oozie-hado-W@count-each-letter OK      job_1342136595052_0006 SUCCEEDED  
-----  
0000001-120711224224630-oozie-hado-W@find-max-letter  RUNNING job_1342136595052_0008  
-----
```

Get info by  
Application ID

Workflow overview

Completed and  
executing tasks

35

# 5: Monitor Running Workflow(s) - Web Interface

<http://localhost:11000/oozie>

The screenshot shows the Oozie Web Console interface in a Mozilla Firefox browser. The address bar displays <http://localhost:11000/oozie/>. The page title is "Oozie Web Console - Mozilla Firefox". The main content area shows a table of workflow jobs. The table has columns for Job Id, Name, Status, Run, User, Group, Created, and Started. The first job is highlighted, and its Job Id is 0000001-120711224224630-oozie-hado-W. The status of this job is "SUCCEEDED".

Job Id	Name	Status	Run	User	Group	Created	Started
1 0000001-120711224224630-oozie-hado-W	most-seen-letter	SUCCEEDED	0	hadoop		Fri, 13 Jul 2012 03:08:16 GMT	Fri, 13 Jul 2012 03:08:16 GMT
2 0000000-120711224224630-oozie-hado-W	most-seen-letter	SUCCEEDED	0	hadoop		Fri, 13 Jul 2012 02:30:22 GMT	Fri, 13 Jul 2012 02:30:22 GMT
3 0000026-120623200723222-oozie-hado-W	most-seen-letter	SUCCEEDED	0	hadoop		Sun, 24 Jun 2012 04:31:58 GMT	Sun, 24 Jun 2012 04:31:58 GMT
4 0000025-120623200723222-oozie-hado-W	most-seen-letter	SUCCEEDED	0	hadoop		Sun, 24 Jun 2012 04:23:32 GMT	Sun, 24 Jun 2012 04:23:32 GMT
5 0000024-120623200723222-oozie-hado-W	most-seen-letter	KILLED	0	hadoop		Sun, 24 Jun 2012 04:20:32 GMT	Sun, 24 Jun 2012 04:20:32 GMT
6 0000023-120623200723222-oozie-hado-W	most-seen-letter	KILLED	0	hadoop		Sun, 24 Jun 2012 04:17:42 GMT	Sun, 24 Jun 2012 04:17:42 GMT
7 0000022-120623200723222-oozie-hado-W	most-seen-letter	SUCCEEDED	0	hadoop		Sun, 24 Jun 2012 04:12:05 GMT	Sun, 24 Jun 2012 04:12:05 GMT

36

# 5: Monitor Running Workflow(s) - Web Interface

The screenshot shows the Oozie Web Console interface in a Mozilla Firefox browser. The address bar displays <http://localhost:11000/oozie/>. The page title is "Oozie Web Console - Mozilla Firefox". The main content area shows the details of a specific workflow job. The job name is "most-seen-letter" and the job ID is "0000001-120711224224630-oozie-hado-W". The "Actions" section shows a table of actions.

Action Id	Name	Type	Status	Transition	StartTime
1 0000001-120711224224630-oozie-hado-W@c	count-each-lett	map-reduce	OK	find-max-letter	Fri, 13 Jul 2012 03:08:16 GMT
2 0000001-120711224224630-oozie-hado-W@fi	find-max-letter	map-reduce	OK	clean-up	Fri, 13 Jul 2012 03:08:43 GMT
3 0000001-120711224224630-oozie-hado-W@c	clean-up	fs	OK	end	Fri, 13 Jul 2012 03:09:08 GMT

Clicking on a particular application/job will bring up list of all the actions; click on each action to get further details

37



## 5: Monitor Running Workflow(s) - Web Interface

The screenshot shows a modal window titled "Action (Name: find-max-letter; JobId: 0000001-120711224224630-oozie-hado-W)". The window has two tabs: "Action Info" and "Action Configuration". The "Action Info" tab is active and displays the following details:

- Name: find-max-letter
- Type: map-reduce
- Transition: clean-up
- Start Time: Fri, 13 Jul 2012 03:08:43 GMT
- End Time: Fri, 13 Jul 2012 03:09:08 GMT
- Status: OK
- Error Code:
- Error Message:
- External ID: job\_1342136595052\_0008
- External Status: SUCCEEDED
- Console URL: [http://localhost:8088/proxy/application\\_1342136595052\\_0008/](http://localhost:8088/proxy/application_1342136595052_0008/)
- Tracker URI: localhost:8021

In the background, a table lists actions with columns for Name, StartTime, and EndTime. A mouse cursor is pointing at the "Console URL" field.

Link to Resource Manager to view details of the job for this particular Oozie Action

This view displays details for a selected action.

38

## 5: Monitor Running Workflow(s) - Web Interface

The screenshot shows a web browser window with the URL [http://localhost:19888/jobhistory/job/job\\_1342136595052\\_0008/](http://localhost:19888/jobhistory/job/job_1342136595052_0008/). The page title is "MapReduce Job job\_1342136595052\_0008". The job details are as follows:

- Job Name: oozie:action:T=map-reduce:W=most-seen-letter:A=find-max-letter:ID=0000001-120711224224630-oozie-hado-W
- User Name: hadoop
- Queue: default
- State: SUCCEEDED
- Uberized: false
- Started: Thu Jul 12 23:08:59 EDT 2012
- Finished: Thu Jul 12 23:09:07 EDT 2012
- Elapsed: 8sec
- Diagnostics:
  - Average Map Time: 3sec
  - Average Reduce Time: 0sec
  - Average Shuffle Time: 3sec
  - Average Merge Time: 3sec

A mouse cursor is pointing at the "State" field.

Oozie assigns a name to each job

Clicking on the "Console Url" from action view will take you to the details of the job for that action

39

## mostSeenLetter-oozieWorkflow Result

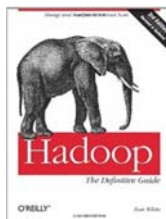
```
$ hdfs dfs -cat /training/playArea/oozieWorkflow/part-r-00000  
t      3711
```

Reminder: This source is in *HadoopSamples* project under  
/src/main/resources/mr/workflows

40

## Oozie Resources

- **Home Page:**
  - <http://incubator.apache.org/oozie/>
  - Quick start, functional specifications for workflows, coordinators, and expression language
- **Mailing Lists**
  - <http://oozie.apache.org/mail-lists.html>
- **Chapter about Oozie**



**Hadoop: The Definitive Guide**

Tom White (Author)

O'Reilly Media; 3rd Edition (May6, 2012)

41



# Wrap-Up

**Customized Java EE Training:** <http://courses.coreservlets.com/>

Hadoop, Java, JSF 2, PrimeFaces, Servlets, JSP, Ajax, jQuery, Spring, Hibernate, RESTful Web Services, Android.  
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

## Summary

- **We learned about**
  - Oozie Features
  - Oozie Installation
- **We learned how to**
  - Implement an Oozie Workflow
  - Deploy and Run Oozie Workflow



# Questions?

More info:

<http://www.coreservlets.com/hadoop-tutorial/> – Hadoop programming tutorial

<http://courses.coreservlets.com/hadoop-training.html> – Customized Hadoop training courses, at public venues or onsite at *your* organization

<http://courses.coreservlets.com/Course-Materials/java.html> – General Java programming tutorial

<http://www.coreservlets.com/java-8-tutorial/> – Java 8 tutorial

<http://www.coreservlets.com/JSF-Tutorial/jsf2/> – JSF 2.2 tutorial

<http://www.coreservlets.com/JSF-Tutorial/primefaces/> – PrimeFaces tutorial

<http://coreservlets.com/> – JSF 2, PrimeFaces, Java 7 or 8, Ajax, jQuery, Hadoop, RESTful Web Services, Android, HTML5, Spring, Hibernate, Servlets, JSP, GWT, and other Java EE training

**Customized Java EE Training: <http://courses.coreservlets.com/>**

Hadoop, Java, JSF 2, PrimeFaces, Servlets, JSP, Ajax, jQuery, Spring, Hibernate, RESTful Web Services, Android.

Developed and taught by well-known author and developer. At public venues or onsite at *your* location.