

大数据Hadoop高薪直通车课程

工作流调度框架Oozie

讲师：轩宇（北风网版权所有）

课程大纲

1

Hadoop 调度框架

2

Oozie 功能架构

3

Oozie 安装部署

4

Oozie 工作流调度

5

Oozie 协作调度

课程大纲

1

Hadoop 调度框架

2

Oozie 功能架构

3

Oozie 安装部署

4

Oozie 工作流调度

5

Oozie 协作调度

Action Nodes

◆ Hive Action

http://archive.cloudera.com/cdh5/cdh/5/oozie-4.0.0-cdh5.3.6/DG_HiveActionExtension.html

◆ Sqoop Action

http://archive.cloudera.com/cdh5/cdh/5/oozie-4.0.0-cdh5.3.6/DG_SqoopActionExtension.html

◆ Shell Action

http://archive.cloudera.com/cdh5/cdh/5/oozie-4.0.0-cdh5.3.6/DG_ShellActionExtension.html

Shell Action

How to run any shell script or perl script or CPP executable

```
<workflow-app xmlns='uri:oozie:workflow:0.3' name='shell-wf'>
  <start to='shell1' />
  <action name='shell1'>
    <shell xmlns="uri:oozie:shell-action:0.1">
      <job-tracker>${jobTracker}</job-tracker>
      <name-node>${nameNode}</name-node>
      <configuration>
        <property>
          <name>mapred.job.queue.name</name>
          <value>${queueName}</value>
        </property>
      </configuration>
      <exec>${EXEC}</exec>
      <argument>A</argument>
      <argument>B</argument>
      <file>${EXEC}#${EXEC}</file> <!--Copy the executable to compute node's current working directory -->
    </shell>
    <ok to="end" />
    <error to="fail" />
  </action>
  <kill name='fail'>
    <message>Script failed, error message[${wf:errorMessage(wf:lastErrorNode())}]</message>
  </kill>
  <end name='end' />
</workflow-app>
```

Shell Action

The corresponding job properties file used to submit Oozie job could be as follows:

```
oozie.wf.application.path=hdfs://localhost:8020/user/kaurul/workflows/script#Ex
#Shell Script to run
EXEC=script.sh
#CPP executable. Executable should be binary compatible to the compute node OS.
#EXEC=hello
#Perl script
#EXEC=script.pl
jobTracker=localhost:8021
nameNode=hdfs://localhost:8020
queueName=default
```

课程大纲

1

Hadoop 调度框架

2

Oozie 功能架构

3

Oozie 安装部署

4

Oozie 工作流调度

5

Oozie 协作调度

Triggering Mechanism

As of now, the Oozie coordinator supports two of the most common triggering mechanisms: **namely time** and **data availability**

Time Trigger

Time-based triggers are easy to explain and resembles the Unix cron utility. In a time-aware coordinator, a workflow is executed at fixed intervals or frequency. A user typically specifies a time trigger in the coordinator using three attributes.

Start time (ST)

Determines when to execute the first instance of the workflow

Frequency (F)

Specifies the interval for the subsequent executions

End time (ET)

Bounds the last execution start time (i.e., no new execution is permitted on or after this time)

In other words, the first execution occurs at the ST and subsequent executions occur at $(ST + F)$, $(ST + 2F)$, $(ST + 3F)$, and so on until the ET is reached.

Data Availability Trigger

Workflow jobs usually process some input data and produce new output data. Therefore, it is very common to hold off the workflow execution until all of the required input data becomes available. For instance, you want to execute a workflow at 1 a.m., but you also want to make sure the required input data is available before the workflow starts. You ideally want the job to wait even past 1 p.m. if any of the input data is missing.

The Oozie coordinator supports a very flexible data dependency-based triggering framework. It is important to note that the concept of data availability-based scheduling is a little more involved than time-based triggering.

时区

◆ 为什么会将地球分为不同时区呢？因为地球总是自西向东自转，东边总比西边先看到太阳，东边的时间也总比西边的早。东边时刻与西边时刻的差值不仅要以時計，而且还要以分和秒来计算。整个地球分为二十四时区，每个时区都有自己的本地时间。在国际无线电通信场合，为了统一起见，使用一个统一的时间，称为通用协调时(UTC, Universal Time Coordinated)。UTC与格林尼治平均时(GMT, Greenwich Mean Time)一样，都与英国伦敦的本地时相同。

CST: 中国标准时间 (China Standard Time)，这个解释可能是针对RedHat Linux。

UTC: 协调世界时，又称世界标准时间，简称UTC，从英文国际时间/法文协调时间"Universal Time Temps Cordonné"而来。中国大陆、香港、澳门、台湾、蒙古国、新加坡、马来西亚、菲律宾、澳洲西部的时间与UTC的时差均为+8，也就是UTC+8。

GMT: 格林尼治标准时间 (旧译格林威治平均时间或格林威治标准时间；英语：Greenwich Mean Time, GMT) 是指位于英国伦敦郊区的皇家格林尼治天文台的标准时间，因为本初子午线被定义在通过那里的经线。

系统时区查看

1: 使用date命令查看时区

```
[root@db-server ~]# date -R  
Sun, 11 Jan 2015 07:10:28 -0800  
[root@db-server ~]#
```

如上RFC 2822 format所示，上面命令输出了-0800表示西八区，是美国旧金山所在的时区，下面表示我们国家的东八区(+0800)

```
[root@lnx01 ~]# date -R  
Sun, 11 Jan 2015 23:06:02 +0800
```

2: 查看clock系统配置文件

```
[root@db-server ~]# more /etc/sysconfig/clock  
ZONE="America/Los_Angeles"  
UTC=true  
ARC=false
```

修改时区

复制相应的时区文件，替换系统时区文件；或者创建链接文件

在 /usr/share/zoneinfo/ 下面有很多时区文件，如下所示，可以复制这些时区文件覆盖 /etc/localtime 文件，或修改符号链接 /etc/localtime 对应的文件

```
[root@db-server ~]# cd /usr/share/zoneinfo/
[root@db-server zoneinfo]# ls
Africa      Chile      Factory   Iceland   Mexico    posix     Universal
America    CST6CDT   GB        Indian    India     Posix/le  UIC
Antarctica  Cuba      GB-Eire   Iran      ISO-3166 PRC       WET
Arctic      EET       GMT       iso3166.tab MET7MDT   PST8PDT   W-SU
Asia        Egypt    GMT0      Israel    Navajo    right     zone.tab
Atlantic    Eire      GMT-0     Jamaica   NZ        ROC       Zulu
Australia   EST       GMT+0     Japan     NZ-CHAT   ROK
Brazil      EST5EDT   green.ic  Kwajalein Pacific    Singapore
Canada      Etc       Hongkong Libya     Poland    Turkey
CET         Europe    HST      MET       Portugal  UCT
[root@db-server zoneinfo]#
```

修改时区

```
[root@db-server ~]# date -R
Mon, 12 Jan 2015 10:42:26 +0800
[root@db-server ~]# cp /usr/share/zoneinfo/America/Los_Angeles
/etc/localtime
cp: overwrite '/etc/localtime'? y
[root@db-server ~]# date -R
Sun, 11 Jan 2015 18:42:49 -0800
[root@db-server ~]#
```

```
[root@db-server ~]#
[root@db-server ~]# date -R
Mon, 12 Jan 2015 10:42:26 +0800
[root@db-server ~]# cp /usr/share/zoneinfo/America/Los_Angeles /etc/localtime
cp: overwrite '/etc/localtime'? y
[root@db-server ~]# date -R
Sun, 11 Jan 2015 18:42:49 -0800
[root@db-server ~]#
```


修改时区

修改链接/etc/localtime的时区文件

```
[root@db-server ~]# ln /usr/share/zoneinfo/Asia/Shanghai /etc/localtime
ln: creating hard link `/etc/localtime' to `/usr/share/zoneinfo/Asia/Shanghai': File exists

[root@db-server ~]# rm /etc/localtime
rm: remove regular file `/etc/localtime'? y

[root@db-server ~]# ln -sf /usr/share/zoneinfo/Asia/Shanghai /etc/localtime

[root@db-server ~]# date -R
Mon, 12 Jan 2015 10:56:10 +0800
```

Oozie 时区问题

oozie默认使用UTC时区，而服务器上可能是CST，建议统一使用GMT+0800。

不要修改oozie-default.xml，无效。在oozie-site.xml中添加：

```
1 <property>
2   <name>oozie.processing.timezone</name>
3   <value>GMT+0800</value>
4 </property>
```

可以用 `oozie info --timezones` 来查看支持的时区

使用GMT+0800后，时间不可以再使用形如 `2014-01-24T13:40Z` 的格式，要使用对应的形如 `2014-01-24T13:40+0800` 的格式

还有一点比较重要，即oozie web console的Time Zone设置要和上述一致，否则你在web console中看到的时间在感官上都是不正确的

`$OOZIE_HOME/oozie-server/webapps/oozie/oozie-console.js`

```
177 function getTimeZone () {
178     Ext.state.Manager.setProvider(new Ext.state.CookieProvider());
179     return Ext.state.Manager.get("TimezoneId", "GMT+0800");
180 }
```


Oozie 时区问题

```
new 1x new 2x oozie-sla-graph.js x
157      xaxis : {
158          show : true,
159          ticks: 12,
160          // labelAngle: -60,
161          mode : "time",
162          timeformat : "%Y-%m-%d %H:%M",
163          //timezone : selectedTimezone
164          timezone: 'GMT+0800'
165      },
```

Oozie Coordinator

All the datetimes used in coordinator applications and job parameters to coordinator applications must be specified in the Oozie processing timezone. If Oozie processing timezone is UTC , the qualifier is **Z** . If Oozie processing time zone is other than UTC , the qualifier must be the GMT offset, (+/-)####.

For example, a datetime in UTC is 2012-08-12T00:00Z, the same datetime in GMT+5:30 is 2012-08-12T05:30+0530 .

For simplicity, the rest of this specification uses UTC datetimes.

If the Oozie processing timezone is UTC , all datetime values are always in UTC down to a minute precision, 'YYYY-MM-DDTHH:mmZ'.

For example 2009-08-10T13:10Z is August 10th 2009 at 13:10 UTC.

If the Oozie processing timezone is a GMT offset GMT(+/-)#### , all datetime values are always in ISO 8601 in the corresponding GMT offset down to a minute precision, 'YYYY-MM-DDTHH:mmGMT(+/-)####'.

For example 2009-08-10T13:10+0530 is August 10th 2009 at 13:10 GMT+0530, India timezone.

Oozie Coordinator

EL Constant	Value	Example
<code>\${coord:minutes(int n)}</code>	<i>n</i>	<code>\${coord:minutes(45)}</code> --> 45
<code>\${coord:hours(int n)}</code>	<i>n * 60</i>	<code>\${coord:hours(3)}</code> --> 180
<code>\${coord:days(int n)}</code>	<i>variable</i>	<code>\${coord:days(2)}</code> --> minutes in 2 full days from the current date
<code>\${coord:months(int n)}</code>	<i>variable</i>	<code>\${coord:months(1)}</code> --> minutes in a 1 full month from the current date
<code>\${cron syntax}</code>	<i>variable</i>	<code>\${0, 10 15 * * 2-6}</code> --> a job that runs every weekday at 3:00pm and 3:10pm UTC time

Cron expressions are comprised of 5 required fields. The fields respectively are described as follows:

Field name	Allowed Values	Allowed Special Characters
Minutes	0-59	, - * /
Hours	0-23	, - * /
Day-of-month	1-31	, - * ? / L W
Month	1-12 or JAN-DEC	, - * /
Day-of-Week	1-7 or SUN-SAT	, - * ? / L #

Oozie Coordinator

Examples:

Cron Expression	Meaning
10 9 * * *	Runs everyday at 9:10am
10,30,45 9 * * *	Runs everyday at 9:10am, 9:30am, and 9:45am
0 * 30 JAN 2-6	Runs at 0 minute of every hour on weekdays and 30th of January
0/20 9-17 * * 2-5	Runs every Mon, Tue, Wed, and Thurs at minutes 0, 20, 40 from 9am to 5pm
1 2 L-3 * *	Runs every third-to-last day of month at 2:01am
1 2 6W 3 ?	Runs on the nearest weekday to March, 6th every year at 2:01am
1 2 * 3 3#2	Runs every second Tuesday of March at 2:01am every year
0 10, 13 * * MON-FRI	Runs every weekday at 10am and 1pm

Oozie Coordinator

```
<coordinator-app name="cron-coord" frequency="0/10 1/2 * * *" start="${start}" end="${end}" timezone="UTC"
  xmlns="uri:oozie:coordinator:0.2">
  <action>
    <workflow>
      <app-path>${workflowAppUri}</app-path>
      <configuration>
        <property>
          <name>jobTracker</name>
          <value>${jobTracker}</value>
        </property>
        <property>
          <name>nameNode</name>
          <value>${nameNode}</value>
        </property>
        <property>
          <name>queueName</name>
          <value>${queueName}</value>
        </property>
      </configuration>
    </workflow>
  </action>
</coordinator-app>
```

本课程版权归北风网所有

欢迎访问我们的官方网站

www.ibeifeng.com