HITACHI
Inspire the Next

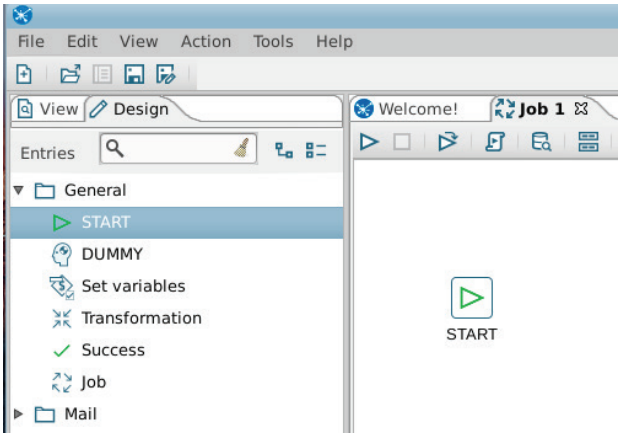
Pentaho MapReduce

Pentaho Data Integration, or PDI, is a comprehensive data integration platform allowing you to access, prepare and derive value from both traditional and big data sources. During this lesson, you will be introduced to Pentaho MapReduce, a powerful alternative to authoring MapReduce jobs that will reduce the technical skills required to use MapReduce, improve productivity and has demonstrable performance advantages over alternative authoring approaches such as Hive, Pig and hand-coded MapReduce jobs. Pentaho MapReduce jobs are designed from the ground up using Pentaho Data Integration's easy-to-use graphical designer, and the resulting jobs leverage the Data Integration engine running in-cluster for maximum performance.

In this lesson, we will re-create the standard Word Count MapReduce example using Pentaho MapReduce. Begin by creating a new Job and adding the 'Start' entry onto the canvas. Jobs in Pentaho Data Integration are used to orchestrate events such as moving files, checking conditions like whether or not a target database table exists, or calling other jobs and transformations. The first thing we want to do in our job is copy the input files containing the words we want to count from our local file system into HDFS.

Step by step with Pentaho:

1. *In PDI, click File - New - Job*
2. *Expand the General tab, drag a 'Start' entry onto the canvas*

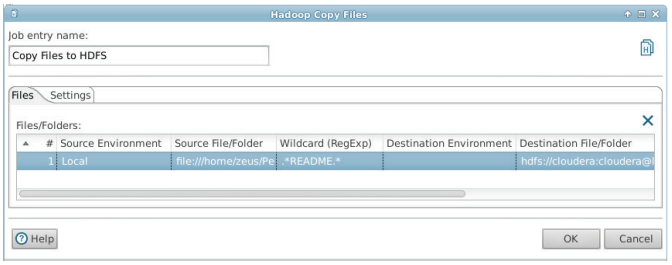


3. *Expand the Big Data tab, drag a 'Hadoop Copy Files' step onto the canvas*
4. *Draw a hop from 'Start' to 'Hadoop Copy Files'*

The Hadoop Copy Files entry makes browsing HDFS as simple as working with your local file system. We'll first specify the folder on our local file system from which we'll copy our Word Count input files. Next, we'll specify the target copy location in HDFS. Finally, we'll define a Wildcard expression to grab all of the files in our source directory that contain the word 'README', and click 'Add' and OK.

Step by step with Pentaho:

1. *Double-click on the 'Hadoop Copy Files' entry to open the job entry editor*
2. *Enter a job entry name of 'Copy Files to HDFS'*
3. *For File/Folder Source enter file:///home/cloudera/pentaho/design-tools/data-integration or the source of the data for your implementation.*
4. *For 'File/Folder destination' copy hdfs://cloudera:cloudera@localhost:8020/wordcount/input or the HDFS directory you are using for your implementation. If you are unsure, please contact your Hadoop systems administrator.*
5. *Copy or enter .*README.* in the 'Wildcard (RegExp)' field*



6. *Click 'OK' to exit the job entry editor*

In the next steps we'll add the Pentaho MapReduce job entry into the flow. Opening the entry editor, you can see that configuring a Pentaho MapReduce job entry is as simple as specifying a Pentaho Data Integration transformation to use as the Mapper, one for the Reducer, and optionally choosing one to act as a Combiner. Since we have not designed our Mapper and Reducer transformations yet, we'll save our job here and pause to go create them.

Step by step with Pentaho:

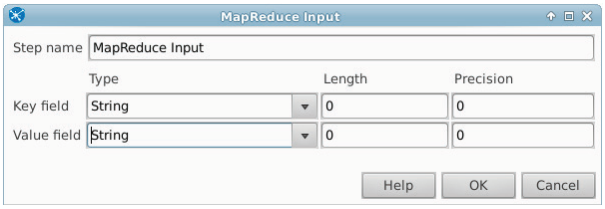
1. *From the Big Data tab drag a 'Pentaho MapReduce' job entry onto the canvas and draw a hop from the' Copy Files to HDFS' step to it.*

2. *Click the 'Save' button, enter 'wordcount-example' as the File name, then click Save*

Next, we will design the transformation to be used as a Mapper in our Word Count example. Any transformation that will be used as a Mapper, Combiner, or Reducer will need to begin with the MapReduce Input step, and end with the MapReduce Output step. These steps are how Hadoop will feed key-value pairs for processing into our transformation, and how the transformation will pass key-value pairs back to Hadoop.

Step by step with Pentaho:

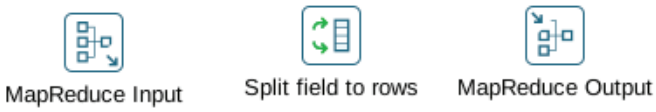
1. *Click File | New | Transformation*
2. *Expand the 'Big Data' folder in the design palate and drag a 'MapReduce Input' step onto the canvas*
3. *Drag a 'MapReduce Output' step onto the canvas*
4. *Double-click the 'MapReduce Input' step and change Type for both Key field and Value field to 'String'.*



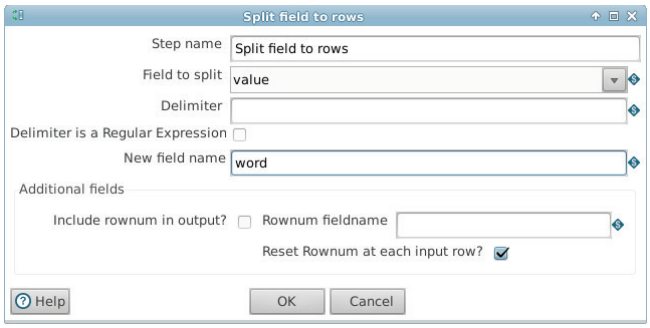
As key-value pairs are fed into this transformation, the first step we want to do is split the strings coming in as values into individual words that we can then use for counting. For this, we'll use the 'Split fields to rows' step. The 'Field to split' will be our 'value' field, we'll split the words based on a SPACE character, and we'll name the resulting field containing the individual words 'word'. This new 'word' field will be used later as our key field for the Reducer part of the Pentaho MapReduce job.

Step by step with Pentaho:

1. *Expand the 'Transform' folder in the design palate, drag a 'Split field to rows' step onto the canvas*



2. *Draw a hop from 'MapReduce Input' to 'Split field to rows' step*
3. *Double-click on the 'Split fields to rows' step to open the step editor*
4. *In the 'Field to split' drop down, select 'value'*
5. *In the 'Delimiter' field, replace the default semi-colon with a SPACE character*
6. *Enter a value of 'word' in the 'New field name' input field*
7. *Click 'OK' to exit the step edito*

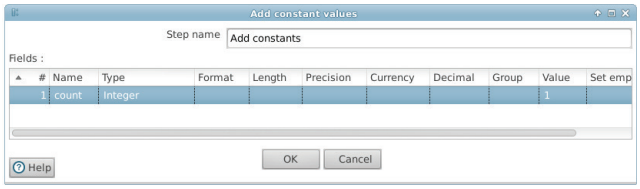


Next in our transformation flow, we'll want to add a new field containing a value of 1 for each individual word extracted. This field will represent our values during the Reducer phase of the Pentaho MapReduce job, used to aggregate the individual instances of a word from our input files. For this, we'll use the 'Add constants' step to create a new field called 'count', with a data type of 'Integer' and a constant value of '1'.

Step by step with Pentaho:

1. *From the 'Transform' folder in the design palate, drag an 'Add constants' step onto the canvas*
2. *Draw a hop from 'Split field to rows' to the 'Add constants' step*

3. *Double-click on the 'Add constants' step to open the step editor*
4. *On line one, enter 'count' as the Name, select 'Integer' as the Type, and a '1' for the Value*



5. *Click 'OK' to exit the step editor*

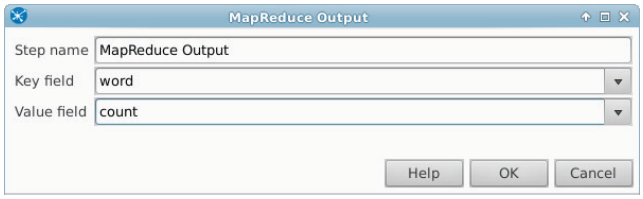
Finally, we'll connect 'Add constants' step to the 'MapReduce Output' step, and configure it to use the new fields 'word' and 'count' as our keys and values to be passed back to Hadoop.

Step by step with Pentaho:

1. *Draw a hop from 'Add constants' to 'MapReduce Output'*

We've now completed a Mapper transformation that will read in key-value pairs, split the incoming values into individual words to be used as our new keys, added a constant of 1for each individual word found that will be used as our values to aggregate during the Reducer phase. Let's save our transformation and move on to designing the Reducer transformation.

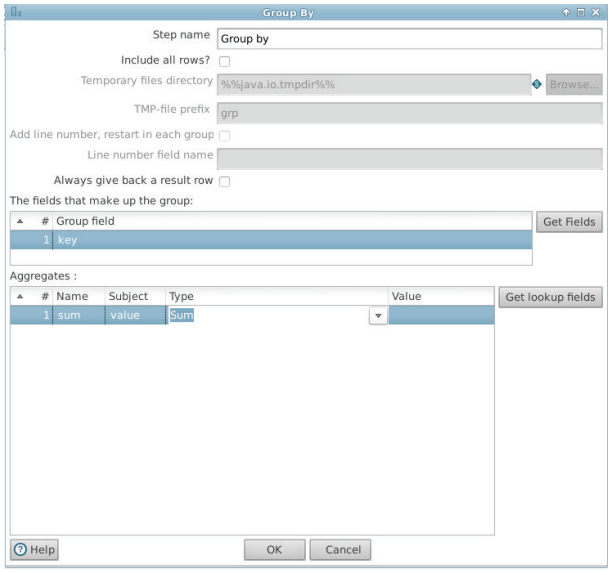
2. *Double-click on 'MapReduce Output' to open the step editor*
3. *Select 'word' as the 'Key field'*
4. *Select 'count' as the 'Value field'*
5. *Click 'OK' to exit the step editor*
6. *Click 'Save'*
7. *Enter 'wordcount-mapper' as the File name, then click 'Save'.*



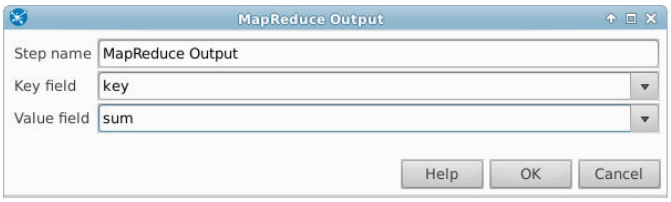
The Reducer transformation will be very simple. We'll create a new transformation and add our 'MapReduce input' and 'MapReduce output' steps. The only business logic we need to add is to aggregate each of our values for the key fields being passed in. For this, we will use the 'Group By' step. Our Group field in this case will be our 'key' field. We'll name the field that will contain our aggregates 'sum', the source column (or Subject) will be the 'value' field, and the type of aggregation we'll perform is 'Sum'. Now this step is configured to group all of the common keys together and create a field called 'sum' containing the total of all values for each instance of a given key, resulting in the total count for how many times a given word appeared in our input files.

Step by step with Pentaho:

1. *Click File|New|Transformation*
2. *Expand the 'Big Data' folder in the design palate, drag a ' MapReduce Input' and a 'MapReduce Output' step onto the canvas*
3. *Expand the 'Statistics' folder in the design palate and drag a 'Group by' step onto the canvas between the Input and Output steps*
4. *Draw a hop from the 'MapReduce Input' step to the ' Group by' step*
5. *Draw a hop from the 'Group by' step to the 'MapReduce Output' step*
6. *Double-click on the 'Group by' step to open the step editor*
7. *In row 1 of 'The fields that make up the group:' table, select 'key' as the Group field*
8. *In row 1 of the 'Aggregates:', enter 'sum' as the Name, select 'value' as the Subject, and select 'Sum' as the Type, then click 'OK' to exit the step editor*



9. Double-click on 'MapReduce Output' to open the step editor
10. Select 'key' as the 'Key field'
11. Select 'sum' as the 'Value field'

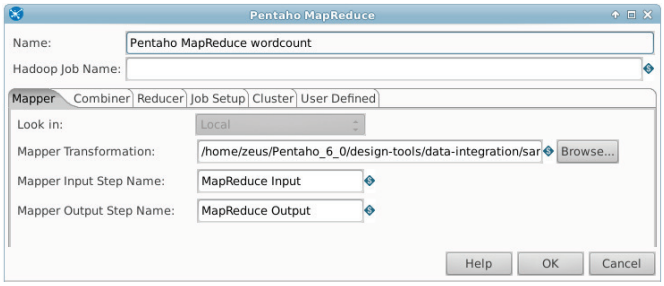


The Reducer transformation is now complete, so we'll return to our Pentaho MapReduce Job and point it at our newly created Mapper and Reducer transformations.

8. Click 'Save', enter 'wordcount-reducer' as the File name, click 'Save'.
9. Click on the 'wordcount-example' (job) tab

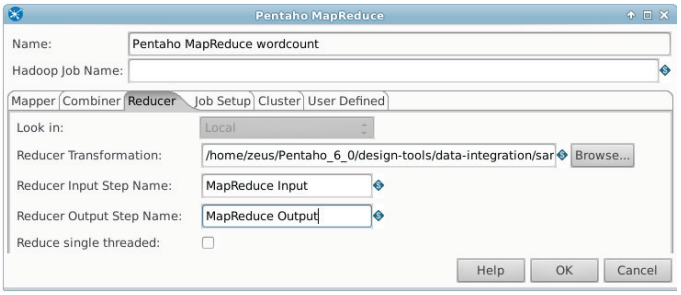
Re-open the Pentaho MapReduce job entry. Provide a name for the Hadoop job. Now we will point the mapper at our 'wordcount-mapper' transformation by selecting the transformation, and specifying the steps within that transformation that represent the Hadoop Input and Output steps.

Step by step with Pentaho:
1. Double-click on the 'Pentaho MapReduce' job entry
2. Enter 'Pentaho MapReduce wordcount'
3. Click on the 'Mapper' tab (may already by selected)
4. Click 'Browse', navigate to and select the 'word-count-mapper.ktr' transformation
5. Enter 'MapReduce Input' as the Mapper Input Step Name
6. Enter 'MapReduce Output' as the Mapper Output Step Name



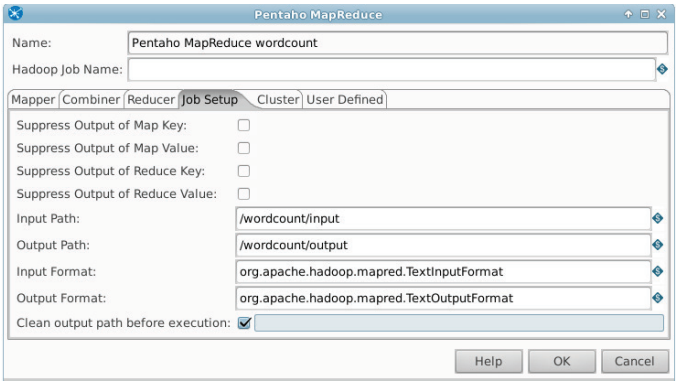
Next, point the reducer to the 'wordcount-reducer' transformation.

1. Click on the 'Reducer' tab
2. Click 'Browse', navigate to and select the 'word-count-reducer.ktr' transformation
3. Enter 'MapReduce Input' as the Mapper Input Step Name
4. Enter 'MapReduce Output' as the Mapper Output Step Name

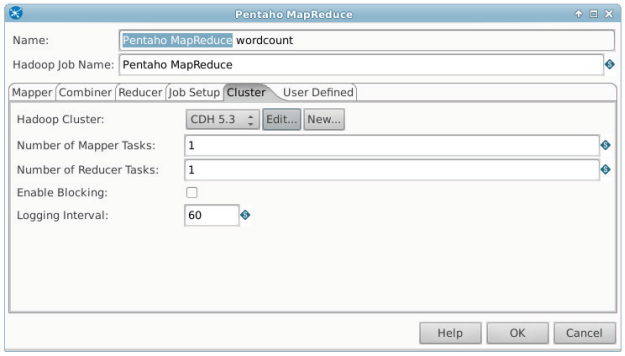


Next, we'll configure our input and output directories for the Hadoop job on the 'Job Setup' tab. The Input Path will be the path we configured to copy our input files to in the Hadoop Copy step, and we'll use '/word-count/output' as the location to output our results. Pentaho MapReduce will support all common formats for input and output data. For this example, we will use the simple TextInputFormat and TextOutputFormat and will select the option to 'Clean output path before execution'

1. Click on the 'Job Setup' tab
2. Enter '/wordcount/input' in the 'Input Path' field
3. Enter '/wordcount/output' in the 'Output Path' field
4. Enter 'org.apache.hadoop.mapred.TextInputFormat' in the 'Input format'
5. Enter 'org.apache.hadoop.mapred.TextOutputFor-mat' in the 'Outputformat'
6. Tick the option to 'Clean output path before execution'.



Finally, we need to point to the Hadoop Cluster on which we want to run the Pentaho MapReduce job. The next step is to enter the cluster details on the Cluster tab. These will depend on how you have set up your environment. If you are unsure of the cluster details, contact your Hadoop systems administrator.



Now we're ready to save our finished job, run it, and view the results. We'll run this job locally, and as the job is executing, we can view the status of the job through feedback gathered from the Hadoop cluster.

1. *Click 'Save' to save your job*
2. *Click the 'Play', then click 'Launch'*
3. *Click on the 'Logging' tab in the Execution results section and pause as the job is running*

Congratulations, you've created your first Pentaho MapReduce job! To find out more information about the powerful Pentaho platform try another lesson or start your free proof of concept with the expertise of a Pentaho sales engineer.

Contact Pentaho at http://www.pentaho.com/contact/.

Hitachi Vantara

Corporate Headquarters
2845 Lafayette Street
Santa Clara, CA 95050-2639 USA
www.HitachiVantara.com | community.HitachiVantara.com

Regional Contact Information
Americas: +1 866 374 5822 or info@hitachivantara.com
Europe, Middle East and Africa: +44 (0) 1753 618000 or info.emea@hitachivantara.com
Asia Pacific: +852 3189 7900 or info.marketing.apac@hitachivantara.com

P-040-A DS February 2018