# Hadoop Job Builder
# 型安全なジョブ設定

2011-02-22
Hadoop Conference
@miyakawa_taku

# 要旨

- Hadoop MapReduce のジョブ設定って面倒です
- 中間データ (map の出力 = reduce の入力) の型を一致させる事がとりわけ面倒です

⇒ 簡潔かつ型安全にジョブを設定する Hadoop Job Builder というライブラリを作りました

# 文書中の単語を数える word-count ジョブの mapper と reducer

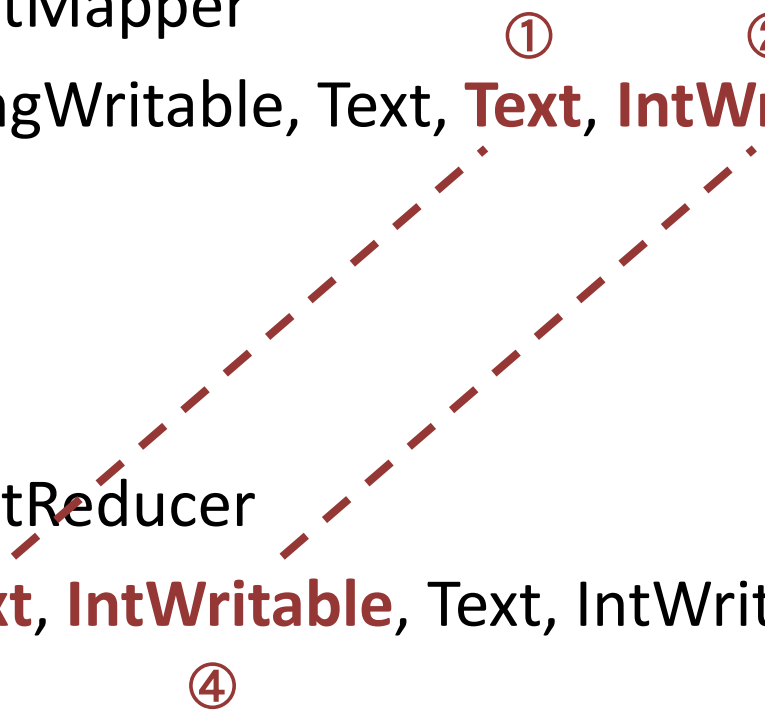
```
public class WordCountMapper ①
  extends Mapper<LongWritable, Text, Text, IntWritable> {
  …
}

public class WordCountReducer ②
  extends Reducer<Text, IntWritable, Text, IntWritable> {
  …
}
```

# 中間データの型は一致させる必要がある
# map の出力 = reduce の入力

```
public class WordCountMapper
   extends Mapper<LongWritable, Text, Text, IntWritable> {
   …
}

public class WordCountReducer
   extends Reducer<Text, IntWritable, Text, IntWritable> {
   …
}
```



# word-count のジョブ設定 こんなに書く

```
Job job = new Job( getConf() );
job.setJobName( "word-count" );
job.setJarByClass( getClass() );
job.setMapperClass( WordCountMapper.class );
job.setMapOutputKeyClass( Text.class );
job.setMapOutputValueClass( IntWritable.class );
job.setReducerClass( WordCountReducer.class );
job.setOutputKeyClass( Text.class );
job.setOutputValueClass( IntWritable.class );
job.setInputFormatClass( TextInputFormat.class );
FileInputFormat.addInputPath( job , new Path( "wordcount/in" ) );
job.setOutputFormatClass( SequenceFileOutputFormat.class );
FileOutputFormat.setOutputPath( job , new Path( "wordcount/out" ) );
```

# やっかいな中間データの設定

```
Job job = new Job( getConf() );
job.setJobName( "word-count" );
job.setJarByClass( getClass() );
job.setMapperClass( WordCountMapper.class );    ①
job.setMapOutputKeyClass( Text.class );         ②
job.setMapOutputValueClass( IntWritable.class ); ③
job.setReducerClass( WordCountReducer.class );  ④
job.setOutputKeyClass( Text.class );
job.setOutputValueClass( IntWritable.class );
job.setInputFormatClass( TextInputFormat.class );
FileInputFormat.addInputPath( job , new Path( "wordcount/in" ) );
job.setOutputFormatClass( SequenceFileOutputFormat.class );
FileOutputFormat.setOutputPath( job , new Path( "wordcount/out" ) );
```

# combiner や partitioner を使うと これらの型も一致させる必要がある

```
Job job = new Job( getConf() );
job.setJobName( "word-count" );
job.setJarByClass( getClass() );
job.setMapperClass( WordCountMapper.class );                    ①
job.setMapOutputKeyClass( Text.class );                         ②
job.setMapOutputValueClass( IntWritable.class );                ③
job.setReducerClass( WordCountReducer.class );                  ④
job.setCombinerClass( WordCountCombiner.class );                ⑤
job.setPartitionerClass( WordCountPartitioner.class );          ⑥
job.setOutputKeyClass( Text.class );
job.setOutputValueClass( IntWritable.class );
job.setInputFormatClass( TextInputFormat.class );
FileInputFormat.addInputPath( job , new Path( "wordcount/in" ) );
job.setOutputFormatClass( SequenceFileOutputFormat.class );
FileOutputFormat.setOutputPath( job , new Path( "wordcount/out" ) );
```

# ちょっとしたパフォーマンスチューニングで

```
public class WordCountMapper
   extends Mapper<LongWritable, Text, Text, IntWritable> {
   …
}

public class WordCountReducer
   extends Reducer<Text, IntWritable, Text, IntWritable> {
   …
}
```

①

②

# 型が一致しなくなると

```
public class WordCountMapper
   extends Mapper<LongWritable, Text, Text, IntWritable> {
   …
}

public class WordCountReducer
   extends Reducer<Text, VIntWritable, Text, VIntWritable> {
   …
}
```

①

②

# コンパイルは通って
# タスクの処理中に実行時エラー

11/02/20 03:07:21 INFO mapred.JobClient: Task Id : attempt_201102200304_0001_r_000000_2, Status : FAILED

java.lang.ClassCastException: org.apache.hadoop.io.IntWritable cannot be cast to org.apache.hadoop.io.VIntWritable

at org.example.WordCountReducer.reduce(WordCountReducer.java:76)

at org.example.WordCountReducer.reduce(WordCountReducer.java:67)

at org.apache.hadoop.mapreduce.Reducer.run(Reducer.java:176)

at org.apache.hadoop.mapred.ReduceTask.runNewReducer(ReduceTask.java:566)

at org.apache.hadoop.mapred.ReduceTask.run(ReduceTask.java:408)

at org.apache.hadoop.mapred.Child.main(Child.java:170)

# MapReduce API は気が利かない

- map, reduce のクラス定義に型情報が含まれているのに、別途わざわざ型を指定するのは冗長
- 型が一致していなかったらコンパイル時にエラーが出てほしい

⇒ 簡潔かつ型安全にジョブを設定したい

# 簡潔かつ型安全にジョブが設定できるライブラリを作りました

```
Job job = JobBuilder.of(
      new WordCountMapper() , new WordCountReducer() )
   .jobName( "word-count" )
   .detectJar()
   .detectKeyValue()
   .inputTextFrom( "wordcount/in" )
   .outputSequenceFileOn( "wordcount/out" )
   .buildJob( getConf() );
```

# 中間データの型をクラス定義から推論して設定
# ⇒ 型が省略できる

```
Job job = JobBuilder.of(
      new WordCountMapper() , new WordCountReducer() )
   .jobName( "word-count" )
   .detectJar()
   .detectKeyValue()
   .inputTextFrom( "wordcount/in" )
   .outputSequenceFileOn( "wordcount/out" )
   .buildJob( getConf() );
```

① ② ③

# 型が一致しないとコンパイルエラー
# ⇒ 型安全

```
Job job = JobBuilder.of(
      new WordCountMapper() , new WordCountReducer() )
    .jobName( "word-count" )
    .detectJar()
    .detectKeyValue()
    .inputTextFrom( "wordcount/in" )
    .outputSequenceFileOn( "wordcount/out" )
    .buildJob( getConf() );
```

① ②

# combiner や partitioner も型安全に設定

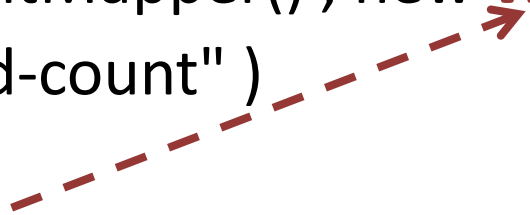
```
Job job = JobBuilder.of(
      new WordCountMapper() , new WordCountReducer() )
    .jobName( "word-count" )
    .detectJar()
    .detectKeyValue()
    .combiner( new WordCountCombiner() )
    .partitioner( new WordCountPartitioner() )
    .inputTextFrom( "wordcount/in" )
    .outputSequenceFileOn( "wordcount/out" )
    .buildJob( getConf() );
```

① ② ③ ④

# 他にも便利

# 出力データの型をクラス定義から推論して設定

```
Job job = JobBuilder.of(
      new WordCountMapper() , new WordCountReducer() )
   .jobName( "word-count" )
   .detectJar()
   .detectKeyValue()
   .inputTextFrom( "wordcount/in" )
   .outputSequenceFileOn( "wordcount/out" )
   .buildJob( getConf() );
```



# 入力元･出力先を簡潔に設定

```
Job job = JobBuilder.of(
      new WordCountMapper() , new WordCountReducer() )
   .jobName( "word-count" )
   .detectJar()
   .detectKeyValue()
   .inputTextFrom( "wordcount/in" )
   .outputSequenceFileOn( "wordcount/out" )
   .buildJob( getConf() );
```

# 分散キャッシュを簡潔に設定

```
Job job = JobBuilder.of(
      new WordCountMapper() , new WordCountReducer() )
   .jobName( "word-count" )
   .detectJar()
   .detectKeyValue()
   .inputTextFrom( "wordcount/in" )
   .outputSequenceFileOn( "wordcount/out" )
   .cacheFileWithSymlink( "/share/dict.txt" , "dict.txt" )
   .buildJob( getConf() );
```

# 総括

- Hadoop Job Builder は簡潔かつ型安全に Hadoop のジョブを設定するライブラリです
- BitBucket に公開しています
  - https://bitbucket.org/miyakawa_taku/hadoop-job-builder/wiki/Home.ja
- 感想ください
  - ⇒ @miyakawa_taku
  - ⇒ BitBucket の issue tracker