Can I use Scala with Apache Beam?

Israel Herraiz <ihr@google.com>
Strategic Cloud Engineer
Google Cloud
@herraiz on Twitter

https://github.com/iht/scio-scala-beam-summit
Scio, the Scala framework

https://github.com/iht/scio-scala-beam-summit
Beam and Scala: Scio

- High level DSL
- Familiarity with Scalding, Spark and Flink
- Functional programming natural fit for data
- Numerical libraries - Breeze, Algebird
- Macros & shapeless for code generation

https://github.com/iht/scio-scala-beam-summit
github.com/spotify/scio
Apache Licence 2.0

https://github.com/iht/scio-scala-beam-summit
Scio Scala API

Apache Beam Java SDK

Batch

Streaming

Cloud Storage

Pub/Sub

BigQuery

Datastore

Bigtable

Scala Libraries

Interactive REPL

Extra features

https://github.com/iht/scio-scala-beam-summit
WordCount

```scala
val sc = ScioContext()
sc.textFile("shakespeare.txt")
  .flatMap { _
    .split("[^a-zA-Z']+")
    .filter(_.nonEmpty)
  }
  .countByValue
  .saveAsTextFile("wordcount.txt")
sc.close()
```

https://github.com/iht/scio-scala-beam-summit
Type safe BigQuery

```scala
@BigQuery.fromFunction("SELECT id, name FROM [users] WHERE ...")
class User // look mom no code!
sc.typedBigQuery[User]().map(u => (u.id, u.name))

@BigQuery.toTable
case class Score(id: String, score: Double)
data.map(kv => Score(kv._1, kv._2)).saveAsTypedBigQuery("table")
```

https://github.com/iht/scio-scala-beam-summit
REPL

Using Scala version 2.11.11 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_121)

Type in expressions to have them evaluated.
Type :help for more information.

Using 'scio-test' as your BigQuery project.
BigQuery client available as 'bq'
Scio context available as 'sc'

scio> _
How would you write Scio yourself?

WordCount example:
- In Beam Java:
  - [https://github.com/nevillelyh/scio-deep-dive/blob/master/src/main/java/WordCount0.java](https://github.com/nevillelyh/scio-deep-dive/blob/master/src/main/java/WordCount0.java)
- Using Beam Java API in a Scala program:
- Wrapping the Java PCollection in a Scala SCollection class:
Learn more

For this workshop: https://github.com/iht/scio-scala-beam-summit

About Scio: https://spotify.github.io/scio/
- Examples: https://spotify.github.io/scio/examples.html

Get started: https://cloud.google.com/free/
Pipeline for our workshop

https://github.com/iht/scio-scala-beam-summit
Each taxi ride sends a message every a few seconds. For each `ride_id`, we will recover the sessions (group and order) all the messages, even if they are coming out of order.

We will calculate:

- Duration of session (event timestamp of most recent message, minus event timestamp of oldest message)
  - Most recent and oldest not in arriving order, but after ordering by event time
- Number of points in the session

[GitHub Link](https://github.com/iht/scio-scala-beam-summit)
The data:

Attributes/metadata in Pubsub:

Pubsub is payload-agnostic

We need attributes to work in event time*

*or we could add timestamps to the data later on before applying the window

https://github.com/iht/scio-scala-beam-summit
Recovering the taxi sessions

```scala
case class PointTaxiRide(  
  ride_id: String,  
  point_idx: Int,  
  latitude: Double,  
  longitude: Double,  
  timestamp: Instant,  
  meter_reading: Double,  
  meter_increment: Double,  
  ride_status: String,  
  passenger_count: Int  
) {

  def toTaxiRide: TaxiRide = {
    case class TaxiRide(  
      ride_id: String,  
      n_points: Int,  
      init: Instant,  
      finish: Option[Instant],  
      total_meter: Double,  
      init_status: String,  
      finish_status: Option[String]  
    )
  }
}
```