Speaker Notes: Scala for Apache Spark

Markus Dale, medale@asymmetrik.com

Jan 2019

Intro, Slides And Code

- Bio:
  - mostly Java, big data with Hadoop
  - big data with Spark, Databricks, Scala
  - Now Asymmetrik - Scala, Spark, Elasticsearch, Akka...
  - Data Engineer
- Slides: https://github.com/medale/scala-spark/blob/master/presentation/ScalaSpark.pdf
- Scala Spark Code Examples: https://github.com/medale/scala-spark
- Also https://github.com/medale/spark-mail

Goals

- Intro to Scala (from Java) to leverage Apache Spark with Scala API
- sbt - build tool
- spark-testing framework for integration testing

Why Scala for Spark?

- Data Engineer - scalable ecosystem of Java/Scala-based tools
- less boilerplate/less typing (Ted Malaska (three big data books on O'Reilly): 50% less than Java)
- strong typing, elegant multi-paradigm language (functional and OO)
- all code runs in executor JVM - no callouts to local Python shell for UDFs/UDAFs
- Baltimore Scala meetup

Java to Scala - Java Main

- semicolons
- get/set JavaBeans convention
- explicit constructor
- static main method

Scala Main One

- Look Ma - no semicolons
- package structure - match directory structure
- Match file name/class name
- object vs. class
    - object - Java static methods, singleton
- no public (default)
- def - method/function declaration
- type declared after variable name
- return type, body (last entry gets returned)
- val - immutable, var - mutable
- class constructor (args none vs. val vs. var)
    - Java get/set: `import scala.reflect.BeanProperty`
    - `@BeanProperty var firstName`

Scala Main One - Output

- println statements
- default class to String - fully qualified class name@...

Scala Main Two - case class

- immutable data structure
- default constructor parameters are `val`
- generates boiler plate code, singleton object

Scala Main Two - javap ScalaMainTwo.class

- javap disassembler (package names removed)
- implements Product (abstract algebraic type), Serializable
    - Serializable important for Spark shuffle!
    - Product
        - productArity
        - productElement(int)
        - productIterator
- static apply factory method/unapply for matching
- copy method
- equals, hashCode, toString (see javap output next)

Want to cover - highlights but have in-depth examples in repo

- Intellij Scala plugin
- object/main method
- case class - Product
- functions - defining a function, anonymous functions
- collections - map, flatMap, filter
- immutability
- implicits - Predef / StringOps / StringLike
- Scala docs
- Spark - RDD, Dataframe, Dataset (Tungsten memory, code gen)
- SparkSession, DataframeReader
- udf
- sbt build - quick overview
- integration testing - spark-testing-base