Spark三种属性配置方式详细说明

随着Spark项目的逐渐成熟,
越来越多的可配置参数被添加到Spark中来。在Spark中提供了三个地方用于配置：

- Spark properties：这个可以控制应用程序的绝大部分属性。并且可以通过 SparkConf 对象或者Java 系统属性进行设置；
- 环境变量(Environment variables)：这个可以分别对每台机器进行相应的设置，比如IP。这个可以在每台机器的 $SPARK_HOME/conf/spark-env.sh 脚本中进行设置；
- 日志：所有的日志相关的属性可以在 log4j.properties 文件中进行设置。

下面对这三种属性设定进行详细的说明。

一、Spark properties

Spark properties可以控制应用程序的绝大部分属性，并且可以分别在每个应用上进行设置。这些属性可以直接在SparkConf对象上设定，该对象可以传递给SparkContext。SparkConf对象允许你去设定一些通用的属性（比如master URL、应用的名称等），这些属性可以传递给set()方法的任意key-value对。如下：

```
val conf = new SparkConf()
             .setMaster("local")
             .setAppName("CountingSheep")
             .set("spark.executor.memory", "1g")
val sc = new SparkContext(conf)
```

动态加载Spark属性

在一些场景中，你可能想避免在代码中将SparkConf对象的属性进行设死；比如，你可能想在不同的master上面或者不同内存容量运行你的应用程序。这就需要你运行程序的时候进行设置，Spark允许你创建一个空的conf对象，如下：

```
val sc = new SparkContext(new SparkConf())
```

然后你可以在运行的时候通过命令行进行一些属性的配置：

```
./bin/spark-submit --name "My app"
                   --master local[4]
                   --conf spark.shuffle.spill=false
                   --conf "spark.executor.extraJavaOptions=-XX:+PrintGCDetails
                           -XX:+PrintGCTimeStamps"
                   myApp.jar
```

Spark shell和 spark-submit工具支持两种方式来动态加载配置属性。第一种是命令行方式，比如--master；spark-submit 工具可以通过--conf标记接收任何的Spark属性。运行 ./bin/spark-submit --help将会显示全部的选项。

./bin/spark-submit 工具也会从 conf/spark-defaults.conf 配置文件中读取配置选项。 在conf/spark-defaults.conf 配置文件中，每行是 key-value
对，中间可以是用空格进行分割，也可以直接用等号进行分割。如下：

```
spark.master            spark://iteblog.com:7077
spark.executor.memory   512m
spark.eventLog.enabled  true
spark.serializer        org.apache.spark.serializer.KryoSerializer
```

每个值将作为一个flags传递到应用中并个SparkConf对象中相应的属性进行合并。通过SparkConf对象配置的属性优先级最高；其次是对spark-submit 或 spark-shell通过flags配置；最后是spark-defaults.conf文件中的配置。

哪里可以查看配置好的Spark属性

在应用程序对应的WEB UI（http://<driver>:4040）上的Environment标签下面将会显示出该应用程序的所有Spark配置选项。在你想确定你的配置是否正确的情况下是非常有用的。需要注意的是，只有显示通过spark-defaults.conf 或SparkConf 进行配置的属性才会在那个页面显示。其他所有没有显示的属性，你可以认为这些属性的值为默认的。

二、环境变量

有很大一部分的Spark设定可以通过环境变量来进行设定。这些环境变量设定在conf/spark-env.sh 脚本文件中（如果你是windows系统，那么这个文件名称是conf/spark-env.cmd）。在Standalone 和 Mesos模式下，这个文件可以设定一些和机器相关的信息（比如hostname）。

需要注意，在刚刚安装的Spark中conf/spark-env.sh文件是不存在的。但是你可以通过复制conf/spark-env.sh.template文件来创建，你的确保这个复制之后的文件是可运行的。

下面的属性是可以在conf/spark-env.sh文件中配置

JAVA_HOME Java的安装目录
PYSPARK_PYTHON Python binary executable to use for PySpark.
SPARK_LOCAL_IP IP address of the machine to bind to.
SPARK_PUBLIC_DNS Hostname your Spark program will advertise to other machines.

对于 standalone 模式的集群除了上面的属性可以配置外，还有很多的属性可以配置，具体我就不说了，自己看文档去。

三、日志配置

Spark用log4j来记录日志。你可以通过配置log4j.properties来设定不同日志的级别、存放位置等。这个文件默认也是不存在的，你可以通过复制log4j.properties.template文件来得到。

在后期文章中，我将逐个的介绍Spark中各个参数的含义。欢迎大家关注。

关于应用程序相关的属性设置解释：《Spark配置属性详解(1)》