翻译规范 - 模版示例 - Spark 概述

原文链接 : http://spark.apache.org/docs/latest/index.html

译文链接 : http://www.apache.wiki/pages/viewpage.action?pageId=2883720

贡献者 : 漫步，那伊抹微笑，ApacheCN，Apache中文网

Apache Spark 是一个快速的、多用途的集群计算系统。在 Java，Scala，Python 和 R 语言以及一个支持常见的图计算的经过优化的引擎中提供了高级 API。它还支持一组丰富的高级工具，包括用于 SQL 和结构化数据处理的 Spark SQL，用于机器学习的 MLlib，用于图形处理的 GraphX、以及 Spark Streaming。

下载

从该项目官网的 下载页面 获取 Spark，该文档用于 Spark 2.0.2 版本。Spark 使用了用于 HDFS 和 YRAN 的 Hadoop client 的库。为了适用于主流的 Hadoop 版本可以下载先前的 package。用户还可以下载 “Hadoop free” binary 并且可以 通过增加 Spark 的 classpath 来与任何的 Hadoop 版本一起运行 Spark。

如果您希望从源码中构建 Spark，请访问 构建 Spark。

Spark 既可以在 Windows 上运行又可以在类似 UNIX 的系统（例如，Linux，Mac OS）上运行。它很容易在一台机器上本地运行 - 您只需要在您的系统 PATH 上安装 Java，或者将 JAVA_HOME 环境变量指向一个 Java 安装目录。

Spark 可运行在 Java 7+，Python 2.6+/3.4 和 R 3.1+ 的环境上。 针对 Scala API，Spark 2.0.1 使用了 Scala 2.11。 您将需要去使用一个可兼容的 Scala 版本（2.11.x）。

运行示例和 Shell

Spark 自带了几个示例程序。 Scala，Java，Python 和 R 的示例在 examples/src/main 目录中。在最顶层的 Spark 目录中使用 bin/run-example <class> [params] 该命令来运行 Java 或者 Scala 中的某个示例程序。（在该例子的底层，调用了 spark-submit 脚本以启动应用程序）。 例如，

```
./bin/run-example SparkPi 10
```

您也可以通过一个改进版的 Scala shell 来运行交互式的 Spark。这是一个来学习该框架比较好的方式。

```
./bin/spark-shell --master local[2]
```

这个 --master 选项可以指定为 分布式集群中的 master URL，或者指定为 local 以使用 1 个线程在本地运行，或者指定为 local[N] 以使用 N 个线程在本地运行 。您应该指定为 local 来启动以便测试。该选项的完整列表，请使用 --help 选项来运行 Spark shell。

Spark 同样支持 Python API。在 Python interpreter（解释器）中运行交互式的 Spark，请使用 bin/pyspark :

```
./bin/pyspark --master local[2]
```

Python 中也提供了应用示例。例如，

```
./bin/spark-submit examples/src/main/python/pi.py 10
```

从 1.4 开始（仅包含了 DataFrames API）Spark 也提供了一个用于实验性的 R API。为了在 R interpreter（解释器）中运行交互式的 Spark，请执行 bin/sparkR :

```
./bin/sparkR --master local[2]
```

R 中也提供了应用示例。例如，

```
./bin/spark-submit examples/src/main/r/dataframe.R
```

在集群上运行

Spark 集群模式概述 说明了在集群上运行的主要的概念。Spark 既可以独立运行，也可以在几个已存在的 Cluster Manager（集群管理器）上运行。它当前提供了几种用于部署的选项：

- Spark Standalone 模式：在私有集群上部署 Spark 最简单的方式。
- Spark on Mesos
- Spark on YARN

快速跳转

编程指南：

- 快速入门：简单的介绍 Spark API，从这里开始！~
- Spark 编程指南：在所有 Spark 支持的语言（Scala，Java，Python，R）中的详细概述。
- 构建在 Spark 之上的模块：
    - Spark Streaming：实时数据流处理。
    - Spark SQL，Datasets，和 DataFrames：支持结构化数据和关系查询。
    - MLlib：内置的机器学习库。
    - GraphX：新一代用于图形处理的 Spark API。

API文档:

- Spark Scala API(Scaladoc)
- Spark Java API(Javadoc)
- Spark Python API(Sphinx)
- Spark R API(Roxygen2)

部署指南:

- 集群模式概述：在集群上运行时概念和组件的概述。
- 提交应用程序：打包和部署应用。
- 部署模式：
    - Amazon EC2：花费大约5分钟的时间让您在EC2上启动一个集群的介绍
    - Spark Standalone 模式：在不依赖第三方 Cluster Manager 的情况下快速的启动一个独立的集群
    - Spark on Mesos：使用 Apache Mesos 来部署一个私有的集群
    - Spark on YARN：在 Hadoop NextGen（YARN）上部署 Spark

其他文件:

- 配置: 通过它的配置系统定制 Spark
- 监控：监控应用程序的运行情况
- 优化指南：性能优化和内存调优的最佳实践
- 作业调度：资源调度和任务调度
- 安全性：Spark 安全性支持
- 硬件配置：集群硬件挑选的建议
- 与其他存储系统的集成：
    - OpenStack Swift
- 构建 Spark：使用 Maven 来构建 Spark
- Contributing to Spark
- Third Party Projects：其它第三方 Spark 项目的支持

外部资源:

- Spark 主页
- Spark Wiki
- Spark 社区 资源，包括当地的聚会
- StackOverflow tag `apache-spark`
- 邮件列表：在这里询问关于 Spark 的问题
- AMP 营地 在加州大学伯克利分校:一系列的训练营,特色和讨论 练习对 Spark,Spark Steaming,Mesos 以及更多。可以免费通过 视频，幻灯片 和 练习 学习。
- 代码示例：更多 Spark 的子文件夹中（Scala，Java，Python，R ）获得。