



# Spark 官方文档翻译

## 在 Yarn 上运行 Spark (v1.2.0)

翻译者 武扬

Spark 官方文档翻译团成员

## 前 言

伴随着大数据相关技术和产业的逐步成熟，继 Hadoop 之后，Spark 技术以集大成的无可比拟的优势，发展迅速，将成为替代 Hadoop 的下一代云计算、大数据核心技术。

Spark 是当今大数据领域最活跃最热门的高效大数据通用计算平台，基于 RDD，Spark 成功的构建起了一体化、多元化的大数据处理体系，在“One Stack to rule them all”思想的引领下，Spark 成功的使用 Spark SQL、Spark Streaming、MLLib、GraphX 近乎完美的解决了大数据中 Batch Processing、Streaming Processing、Ad-hoc Query 等三大核心问题，更为美妙的是在 Spark 中 Spark SQL、Spark Streaming、MLLib、GraphX 四大子框架和库之间可以无缝的共享数据和操作，这是当今任何大数据平台都无可匹敌的优势。

在实际的生产环境中，世界上已经出现很多一千个以上节点的 Spark 集群，以 eBay 为例，eBay 的 Spark 集群节点已经超过 2000 个，Yahoo！等公司也在大规模的使用 Spark，国内的淘宝、腾讯、百度、网易、京东、华为、大众点评、优酷土豆等也在生产环境下深度使用 Spark。2014 Spark Summit 上的信息，Spark 已经获得世界 20 家顶级公司的支持，这些公司中包括 Intel、IBM 等，同时更重要的是包括了最大的四个 Hadoop 发行商，都提供了对 Spark 非常强有力的支持。

与 Spark 火爆程度形成鲜明对比的是 Spark 人才的严重稀缺，这一情况在中国尤其严重，这种人才的稀缺，一方面是由于 Spark 技术在 2013、2014 年才在国内的一些大型企业里面被逐步应用，另一方面是由于匮乏 Spark 相关的中文资料和系统化的培训。为此，Spark 亚太研究院和 51CTO 联合推出了“Spark 亚太研究院决胜大数据时代 100 期公益大讲堂”，来推动 Spark 技术在国内的普及及落地。

具体视频信息请参考 [http://edu.51cto.com/course/course\\_id-1659.html](http://edu.51cto.com/course/course_id-1659.html)

与此同时，为了向 Spark 学习者提供更为丰富的学习资料，Spark 亚太研究院去年 8 月发起并号召，结合网络社区的力量构建了 Spark 中文文档专家翻译团队，翻译了 Spark 中文文档 V1.1.0 版本。2014 年 12 月，Spark 官方团队发布了 Spark 1.2.0 版本，为了让学习者了解到最新的内容，Spark 中文文档专家翻译团队又对 Spark 1.2.0 版本进行了部分更新，在此，我谨代表 Spark 亚太研究院及广大 Spark 学习爱好者向专家翻译团队所有成员热情而专业的工作致以深刻的敬意！

当然，作为相对系统的 Spark 中文文档，不足之处在所难免，大家有任何建议或者意见都可以发邮件到 [marketing@sparkinchina.com](mailto:marketing@sparkinchina.com)；同时如果您想加入 Spark 中文

文档翻译团队，也请发邮件到 [marketing@sparkinchina.com](mailto:marketing@sparkinchina.com) 进行申请；Spark 中文文档的翻译是一个持续更新的、不断版本迭代的过程，我们会尽全力给大家提供更高质量的 Spark 中文文档翻译。

最后，也是最重要的，请允许我荣幸的介绍一下我们的 Spark 中文文档 1.2.0 版本翻译的专家团队成員，他们分别是（排名不分先后）：

- ▶ 傅智勇,《快速开始(v1.2.0)》
- ▶ 王宇舟,《Spark 机器学习库 (v1.2.0)》
- ▶ 武扬,《在 Yarn 上运行 Spark (v1.2.0)》《Spark 调优(v1.2.0)》
- ▶ 徐骄,《Spark 配置(v1.2.0)》《Spark 作业调度(v1.2.0)》
- ▶ 蔡立宇,《Bagel 编程指南(v1.2.0)》
- ▶ harli,《Spark 编程指南 (v1.2.0)》
- ▶ 韩保礼,《Spark SQL 编程指南(v1.2.0)》
- ▶ 李丹丹,《文档首页(v1.2.0)》
- ▶ 李军,《Spark 实时流处理编程指南(v1.2.0)》
- ▶ 俞杭军,《使用 Maven 编译 Spark(v1.2.0)》
- ▶ 王之,《给 Spark 提交代码(v1.2.0)》
- ▶ Ernest,《集群模式概览(v1.2.0)》《监控与相关工具(v1.2.0)》《提交应用程序(v1.2.0)》

Life is short, You need Spark!

Spark 亚太研究院院长 王家林  
2015 年 2 月

## Spark 亚太研究院决胜大数据时代 100 期公益大讲堂

### 简 介

作为下一代云计算的核心技术,Spark 性能超 Hadoop 百倍,算法实现仅有其 1/10 或 1/100,是可以革命 Hadoop 的目前唯一替代者,能够做 Hadoop 做的一切事情,同时速度比 Hadoop 快了 100 倍以上。目前 Spark 已经构建了自己的整个大数据处理生态系统,国外一些大型互联网公司已经部署了 Spark。甚至连 Hadoop 的早期主要贡献者 Yahoo 现在也在多个项目中部署使用 Spark。国内的淘宝、优酷土豆、网易、Baidu、腾讯、皮皮网等已经使用 Spark 技术用于自己的商业生产系统中,国内外的应用开始越来越广泛。Spark 正在逐渐走向成熟,并在这个领域扮演更加重要的角色,刚刚结束的 2014 Spark Summit 上的信息,Spark 已经获得世界 20 家顶级公司的支持,这些公司中包括 Intel、IBM 等,同时更重要的是包括了最大的四个 Hadoop 发行商都提供了对非常强有力的支持 Spark 的支持。

鉴于 Spark 的巨大价值和潜力,同时由于国内极度缺乏 Spark 人才,Spark 亚太研究院在完成了对 Spark 源码的彻底研究的同时,不断在实际环境中使用 Spark 的各种特性的基础之上,推出了 Spark 亚太研究院决胜大数据时代 100 期公益大讲堂,希望能够帮助大家了解 Spark 的技术。同时,对 Spark 人才培养有进一步需求的企业和个人,我们将以公开课和企业内训的方式,来帮助大家进行 Spark 技能的提升。同样,我们也为企业提供一体化的顾问式服务及 Spark 一站式项目解决方案和实施方案。

Spark 亚太研究院决胜大数据时代 100 期公益大讲堂是国内第一个 Spark 课程免费线上讲座,每周一期,从 7 月份起,每周四晚 20:00-21:30,与大家不见不散!老师将就 Spark 内核剖析、源码解读、性能优化及商业实战案例等精彩内容与大家分享,干货不容错过!

时间:从 7 月份起,每周一期,每周四晚 20:00-21:30

形式:腾讯课堂在线直播

学习条件:对云计算大数据感兴趣的技术人员

课程学习地址:[http://edu.51cto.com/course/course\\_id-1659.html](http://edu.51cto.com/course/course_id-1659.html)

# 在 Yarn 上运行 Spark (v1.2.0)

( 翻译者：武扬 )

Running Spark on YARN ,

原文档链接：<http://spark.apache.org/docs/latest/running-on-yarn.html>

## 目录

1. Running Spark on YARN.....	6
1.1 准备工作 .....	6
1.2 配置.....	6
1.2.1 环境变量.....	6
1.2.2 Spark 属性.....	6
2. 在 YARN 启动 Spark 应用.....	8
2.1 添加其他的 Jar 文件 .....	9
2.2 调试应用 .....	10
3. 注意事项 .....	11

# 1. Running Spark on YARN

自 Spark 版本 0.6.0 起, Spark 支持在 [YARN \(Hadoop NextGen\)](#) 上运行, 并且在后续版本中不断改进

## 1.1 准备工作

使用 Spark-on-YARN 需要支持 YARN 的 Spark 预编译版本。预编译版本可以从 Spark 网站中下载。如需自行编译, 请参考 [Maven 编译指引](#)。

## 1.2 配置

Spark On YARN 模式的大部分配置与其他模式相同。参见[配置页面](#)可以看到相关信息。下面的内容将是在 Spark on YARN 模式下的专用配置。

### 1.2.1 环境变量

- SPARK\_YARN\_USER\_ENV, Spark 进程在 YARN 中启动, 需要添加这些环境变量。可以用逗号分隔这些变量, 就像这样  
SPARK\_YARN\_USER\_ENV="JAVA\_HOME=/jdk64,FOO=bar".

### 1.2.2 Spark 属性

属性名	默认值	含义
spark.yarn.applicationMaster.waitTries	10	设置 ApplicationMaster 等待 Spark master 开始工作的次数, 即 SparkContext 可以初始化的重试次数
spark.yarn.submit.file.replication	3	应用向 HDFS 上传文件的副本数。其中包括 Spark 自身的 jar 文件, app 的 jar 文件以及其他分布式存储的文件、压缩包
spark.yarn.preserve.staging.files	false	设置是否将本次使用的文件( Spark 的 jar 文件, app 的 jar 文件, 分布式的文件)在任务完成后删除
spark.yarn.scheduler.	5000	Spark application master ( Spark 应用执

heartbeat.interval-ms		行的主导进程 ) 与 YARN ResourceManager ( YARN 的资源管理器 ) 间的心跳间隔, 以毫秒(ms)为单位
spark.yarn.max.executor.failures	2*executor 数量 , 最小值为 3	在任务失败前, executor ( 任务的执行进程 ) 的最大失败次数
spark.yarn.historyServer.address	(none)	Spark 历史服务器的地址 ( 例如 : host.com:18080 ) , 不要包含协议头 ( http:// ) 。由于这是一个可选的服务, 所以默认没有设置。当 Spark 应用执行完成的时候, YARN ResourceManager 可以通过这个地址将资源管理器的界面与 Spark 历史服务器的界面链接。
spark.yarn.dist.archives	(无)	将要解压到每一个 executor 工作目录下的压缩包, 用逗号分割。
spark.yarn.dist.files	(none)	将要放置到每个 executor 工作目录下的文件, 用逗号分隔。
spark.yarn.executor.memoryOverhead	executor 内存 *0.07 , 最小为 384	每个 executor 可以分配的非堆存储 ( off-heap ) 内存, 单位为 ( MB ) 。这些内存用于如 VM 额外开销, 常量池字符, 以及其他本地额外开销等。这会随着 executor 的大小增长, 通常为 6-10%
spark.yarn.driver.memoryOverhead	driver 内存 *0.07, 最小为 384	每个 driver( Spark 任务的数据驱动进程 ) 可以分配的非堆存储 ( off-heap ) 内存, 单位为 ( MB ) 。这些内存用于如 VM 额外开销, 常量池字符, 以及其他本地额外开销等。这会随着 container 的大小增长, 通常为 6-10%
spark.yarn.queue	默认	任务上传到的 YARN 队列的名字
spark.yarn.jar	(无)	如果想改变默认的位置, 可以指定 Spark jar 文件的位置。默认的, Spark on YARN 会使用安装在本地的 Spark jar 文件, 但是 Spark jar 文件也可以放在 HDFS 可以访问到的任意位置。YARN 可以将他缓存在节点上, 这样就不必在应用每次运行的时候再分发。可以通过设置这个配置如 "hdfs:///some/path" 来指定 HDFS 上的 jar 文件。



spark.yarn.access.namenodes	(无)	Spark 应用需要访问的 HDFS 安全节点的列表。例如， `spark.yarn.access.namenodes=hdfs://nn1.com:8032,hdfs://nn2.com:8032`。这个 Spark 应用必须访问列表中指定的 namenode 并且 Kerberos 必须正确的设置好以能够访问他们（无论是同域或者信任域）。Spark 获取到每个 namenode 的安全令牌，这样 Spark 应用可以访问那些远程 HDFS 的集群。
spark.yarn.appMasterEnv.[EnvironmentVariableName]	(无)	将环境变量通过 EnvironmentVariableName 添加到在 YARN 上启动的 Application Master 进程。用户可以指定多个这样的配置来设置多个环境变量。在 yarn-cluster 模式中，这个可以控制 SPARK driver 的环境变量，如果是 yarn-client 模式这只能控制 executor 启动的环境变量。
spark.yarn.containerLauncherMaxThreads	25	由 application master 启动的 executor 容器的最大线程数

Spark on YARN 模式会默认使用本地安装的 Spark 自身的 jar 文件，但是也可以把 jar 文件放在全局可读的 HDFS 上，这样可以让 YARN 将他缓存到节点上，运行应用时就无需每次都分发到每个节点。用下面的方法可以指定 JAR 文件在 HDFS 的位置 `export SPARK_JAR=hdfs://some/path`。

## 2. 在 YARN 启动 Spark 应用

对于 Hadoop 集群，请确保 HADOOP\_CONF\_DIR 或者 YARN\_CONF\_DIR 指向包含有对应（客户端）配置文件的目录。这些配置会被写入分布式文件系统，YARN ResourceManager 会连接他们。

在 YARN 中有两种部署模式用于启动 Spark 应用。在 yarn-cluster 模式中，Spark 的 driver 会在 application master 进程中运行，受到 YARN 的管理，这样客户端程序可以在应用初始化后脱离。而在 yarn-client 模式中，驱动运行在客户端进程中，application master 仅用于向 YARN 请求资源。

不像 Spark standalone 或者 Mesos 的模式那样，主机地址由参数 “master” 指定。在 YARN 模式中，资源管理器的地址由 Hadoop 的配置文件中指定。因此，主机的参数只要像 “yarn-client” 或者 “yarn-cluster” 这样就可以。

在 yarn-cluster 模式下启动应用：



```
./bin/spark-submit --class path.to.your.Class --master yarn-cluster [options]
<app jar> [app options]

./bin/spark-submit --class path.to.your.Class (类路径) --master yarn-cluster
[options (可选参数)] <app jar> (jar 文件路径) [app options (应用参数)]
```

例如：

```
$ ./bin/spark-submit --class org.apache.spark.examples.SparkPi \
--master yarn-cluster \
--num-executors 3 \
--driver-memory 4g \
--executor-memory 2g \
--executor-cores 1 \
--queue thequeue \
lib/spark-examples*.jar \
10
```

以上展示了使用默认 Application Master 来启动 YARN 客户端。SparkPi 将作为 Application Master 的一个子线程运行。客户端会定期向 Application Master 进行轮询，更新并且在控制台中显示状态。一旦应用运行完成，客户端便会退出。如何查看 driver 以及 executor 的日志请参考“监视日志”章节。

相似的，将“yarn-cluster”改为“yarn-client”就可以在 yarn-client 模式下启动 Spark 应用。运行一个 spark-shell：

```
$ ./bin/spark-shell --master yarn-client
```

## 2.1 添加其他的 Jar 文件

在 yarn-cluster 模式下，driver 与客户端可能运行在不同的机器上，所以 SparkContext.addJar 参数指定的相对于客户端为本地的文件并不能直接使用。用启动参数--jars 来指定可以让那客户端可以在 SparkContext.addJar 中生效。

```
$ ./bin/spark-submit --class my.main.Class \
--master yarn-cluster \
--jars my-other-jar.jar,my-other-other-jar.jar
my-main-jar.jar
```

```
app_arg1 app_arg2
```

## 2.2 调试应用

在 YARN 的术语中,executors 和 application masters 运行在 “container( 容器 )” 中。 在应用完成运行后, YARN 有两种处理 container 日志的模式。如果开启了日志聚合( 通过 yarn.log-aggregation-enable 配置 ), container 的日志会复制到 HDFS 中, 并且删除机器上的本地之日。这样可以通过 “yarn logs” 命令在集群中的任意地方进行查看。

```
yarn logs -applicationId <app ID>
```

运行如上命令会打印出指定应用所有 container 的日志内容。

如果日志聚合是关闭的, 那么日志会保存在每一台机器的目录 YARN\_APP\_LOGS\_DIR 下, 通常根据 Hadoop 的版本和安装位置配置为/tmp/logs 或者 \$HADOOP\_HOME/logs/userlogs。查看某个 container 的日志需要在对应主机的那些目录下进行。子目录会根据 application ID 以及 container ID 来确定。

为了回顾之前的 container 启动环境, 通过增加 yarn.nodemanager.delete.debug-delay-sec 的值(例如 36000), 可以在 yarn.nodemanager.local-dirs 目录下查看那些节点下启动的 container 的应用的缓存记录。这个目录包含启动脚本, Jar 文件以及启动每个 container 所需的所有环境变量。这个过程有助于调试, 尤其是那些 classpath 相关的问题。( 需要注意的是, 开启这个功能需要管理员权限在集群中进行设置, 并且要重启所有的 node managers。因此并不适用于托管的集群。

如果为 application master 或者 executor 使用自定义的 log4j 配置, 可以通过下面两个设置进行:

- 可以用 spark-submit 上传自定义的 log4j.properties , 通过将文件列表用 “-files” 与应用一起上传。
- 增加 “-Dlog4j.configuration=” 到 “spark.driver.extraJavaOptions” ( 对于 driver ) 或者 “spark.executor.extraJavaOptions” ( 对于 executor )。 需要注意的是, 如果使用一个文件, 那么 “file:” 这个文件协议头必须指定, 每个节点上都要放置该文件。

注意, 对于上面第一个方法, executor 和 application master 会共享同一个 log4j 的配置。这样如果他们运行在同一个节点可能会引起问题。( 例如试图去写同一个日志文件 )。

如果你需要指定 log 文件的目录来让 YARN 聚合和显示, 使用 `"{spark.yarn.app.container.log.dir}/spark.log`. 对于流计算应用, 配置 `RollingFileAppender` 并且将文件路径设置到 YARN 的 log 目录可以避免由于巨大的日志文件导致的磁盘过载, 并且可以使用 YARN 的 log 工具来查看他们

### 3. 注意事项

- 在 Hadoop 2.2 之前, YARN 并不支持 container 资源指定核心数。因此, 对于早期版本, 命令行参数指定的核心数量并不会传给 YARN。哪个核被请求将由使用的调度器以及调度器的配置决定。
- Spark executor 使用的本地目录由 YARN 的配置确定 (Hadoop YARN 配置 `yarn.nodemanager.local-dirs`)。即使用户指定了 `spark.local.dir` 也会被忽略。
- 与 Hadoop 类似, 参数选项 `--files` 和 `--archives` 支持通过 `#` 指定文件名称。例如我们可以指定: `--files localtest.txt#appSees.txt`, 这样会把本地文件 `localtest.txt` 上传到 HDFS, 但是仍然会通过文件名 `appSees.txt` 链接, 在 YARN 中运行是, 应用可以使用文件名 `appSees.txt` 来表示他。
- 在 `yarn-cluster` 模式下, 如果你使用本地文件, 参数选项 `--jars` 允许使用 `SparkContext.addJar` 的功能。如果你使用 HDFS, HTTP, HTTPS, 或者 FTP 文件的时候并不需要使用。

## ■ Spark 亚太研究院

Spark 亚太研究院是中国最专业的一站式大数据 Spark 解决方案供应商和高品质大数据企业级完整培训与服务供应商，以帮助企业规划、架构、部署、开发、培训和使用 Spark 为核心，同时提供 Spark 源码研究和应用技术训练。针对具体 Spark 项目，提供完整而彻底的解决方案。包括 Spark 一站式项目解决方案、Spark 一站式项目实施方案及 Spark 一体化顾问服务。

官网：[www.sparkinchina.com](http://www.sparkinchina.com)

## ■ 视频课程：

### 《大数据 Spark 实战高手之路》 国内第一个 Spark 视频系列课程

从零起步，分阶段无任何障碍逐步掌握大数据统一计算平台 Spark，从 Spark 框架编写和开发语言 Scala 开始，到 Spark 企业级开发，再到 Spark 框架源码解析、Spark 与 Hadoop 的融合、商业案例和企业面试，一次性彻底掌握 Spark，成为云计算大数据时代的幸运儿和弄潮儿，笑傲大数据职场和人生！

- ▶ 第一阶段：熟练的掌握 Scala 语言  
课程学习地址：<http://edu.51cto.com/pack/view/id-124.html>
- ▶ 第二阶段：精通 Spark 平台本身提供给开发者 API  
课程学习地址：<http://edu.51cto.com/pack/view/id-146.html>
- ▶ 第三阶段：精通 Spark 内核  
课程学习地址：<http://edu.51cto.com/pack/view/id-148.html>
- ▶ 第四阶段：掌握基于 Spark 上的核心框架的使用  
课程学习地址：<http://edu.51cto.com/pack/view/id-149.html>
- ▶ 第五阶段：商业级别大数据中心黄金组合：Hadoop+ Spark  
课程学习地址：<http://edu.51cto.com/pack/view/id-150.html>
- ▶ 第六阶段：Spark 源码完整解析和系统定制  
课程学习地址：<http://edu.51cto.com/pack/view/id-151.html>

## ■ 图书：

### 《大数据 spark 企业级实战》

京东购买官网：<http://item.jd.com/11622851.html>

当当购买官网：<http://product.dangdang.com/23631792.html>

亚马逊购买官网：<http://www.amazon.cn/dp/B00RMD8KI2/>



咨询电话：4006-998-758

QQ 交流群：1 群：317540673 ( 已满 )

2 群：297931500 ( 已满 )

3 群：317176983

4 群：324099250



微信公众号：spark-china