学习记录

harli

2014/12/8

学习过程中的一些信息，待整理……

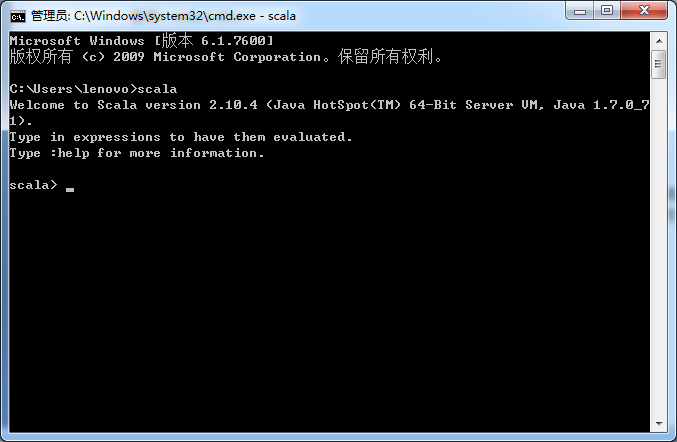
环境搭建

# scala

**版本：**scala 2.10.x

**下载：**<http://www.scala-lang.org/files/archive/scala-2.10.4.msi>

**验证：**安装完成后，在命令行窗口(cmd)输入scala：



# git

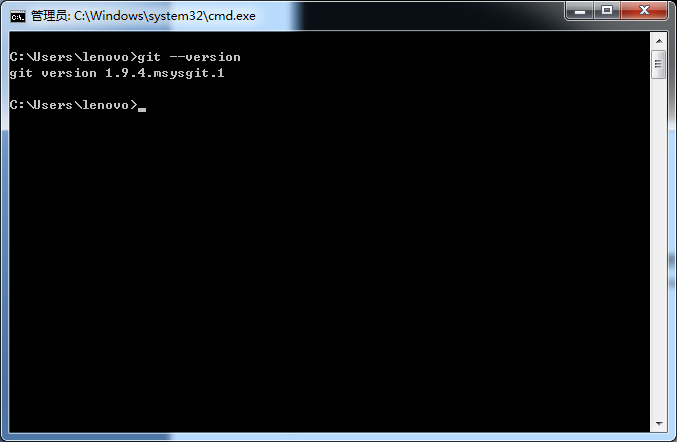
**安装：**

1. 安装支持Windows的Git：此处使用Github页面上的windows的安装版本（GitHubSetup.exe）。
2. 设置环境变量：将git.exe所在的bin目录添加到Path环境变量，如：安装目录\GitHub\PortableGit\_....\bin；

说明：我在安装时，path配置没有自动设置，需手动设置，根据GitHub所在路径，搜索git.exe文件，可以找到对应的bin路径，将该bin路径设置到path中即可。

**验证：**

1. 运行GitHub、Git Shell木有问题！
2. 命令行窗口(cmd)运行：git –version，显示版本信息，安装、配置路径成功！



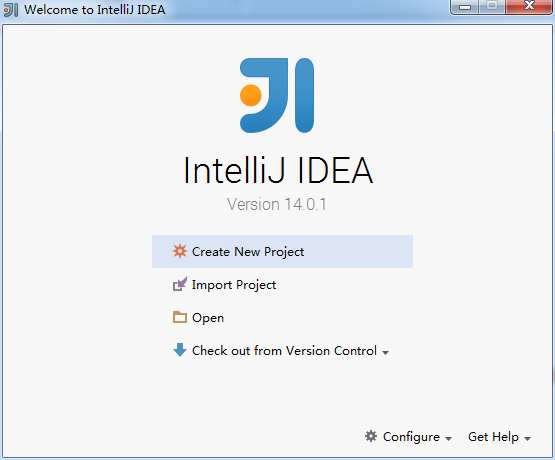
# Intellij IDEA

**要求：**需要安装jdk，git。

Spark相关：scala 2.10.x (注意版本)。

**安装：**

1. 下载最新免费的商业版本 IDEA -14.0.1 ： <http://www.jetbrains.com/idea/download/>；
2. 安装后，启动IDEA，安装scala插件：依次选择“Configure”–> “Plugins”–> “Browse repositories”，输入scala，然后安装即可；



初始界面，在右下角Configure里面。

补充：也可以在相应界面上直接下载scala插件，放到IDEA对应插件目录下， 然后重启IDEA。

# Apache spark的环境搭建

## 源码下载

从github获得spark源码：

git clone <https://github.com/apache/spark.git>

## IntelliJ IDEA

### **搭建Spark源码阅读环境（需要联网）**

官方网站建议用sbt-idea插件建立IntelliJ IDEA的工程。

1. 第一种方法：依次选择“import project”–> 选择spark所在目录 –> “SBT”，之后intellij会自动识别SBT文件，并下载依赖的外部jar包，整个流程用时非常长，取决于机器的网络环境（不建议在windows下操作，可能遇到各种问题），一般需花费几十分钟到几个小时。

注意，下载过程会用到git，因此应该事先安装了git。

1. 第二种方法：首先在linux操作系统上生成intellij项目文件，然后在intellij IDEA中直接通过“Open Project”打开项目。

在linux上生成intellij项目文件的方法（需要安装git，不需要安装scala，sbt会自动下载）：在spark源代码根目录下，输入sbt/sbt gen-idea。

注：如果你在windows下阅读源代码，建议先在linux下生成项目文件，然后导入到windows中的intellij IDEA中。

## **搭建Spark开发环境**

### 应用程序的创建

1. 创建工程：在intellij IDEA中创建scala project；
2. 导入编译的spark依赖包：IDEA中，选择 “File”–> “project structure” –> “Libraries”，选择“+”，添加该spark-hadoop的jar包，如

注：spark应用程序只需要导入spark-hadoop对应的jar依赖包（如spark-1.1.0-hadoop-2.2.0.bin.tgz），不需要其他库。

另外，Scala插件安装之后，IDE会自动识别scala库，如果不识别，再手动导入。

### intellij IDEA调试、运行应用程序

编写完应用程序之后，可以在IDEA上直接调试、运行。

应用程序运行的入口脚本spark-submit.sh，对应SparkSubmit类，可以知道各种模式下应用程序执行方式。

……

1. Local模式下调试、运行

编写完scala程序后，可以在intellij中，以local模式进行调试、运行。

点击“Run”–> “Run Configurations”，设置Program arguments为“local”，表示将该参数传递给main函数，如下图所示，之后点击“Run”–> “Run”运行程序即可。

1. 集群模式下调试、运行
2. Client-cluster两种方式提交、调试等比较

**Driver的提交与调试：**

Client：提交点运行，因此driver的代码应该可以通过SparkSubmit类提交运行、跟踪调试等；

Cluster：在集群中运行，因此需要针对driver类设置调试的OPTS信息，然后通过SparkSubmit类提交后，启动driver代码的跟踪调试；

**Master调试**：打开调试的OPTS信息，针对资源管理调度方面的调试；

**Worker调试**：类似于Master；

**Executor的进程调试**：有待进一步分析 —— 确认下启动进程时，是否已经提供了对应的OPTS设置？

### 命令行提交应用程序

1. 导出jar包

选择“File”–> “Project Structure” –> “Artifact”，选择“+”–> “Jar” –> “From Modules with dependencies”，选择main函数，并在弹出框中选择输出jar位置，并选择“OK”。

最后依次选择“Build”–> “Build Artifact”编译生成jar包。

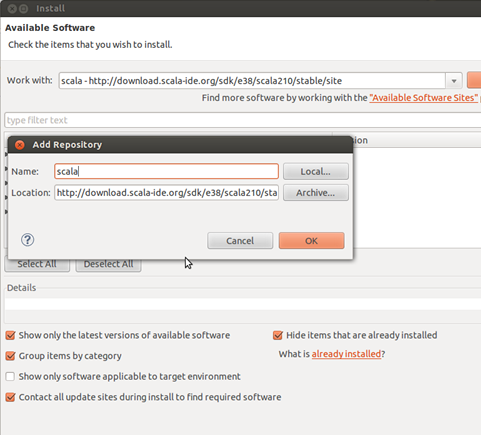
1. 命令行提交

### IntelliJ IDEA中编译调试spark的程序

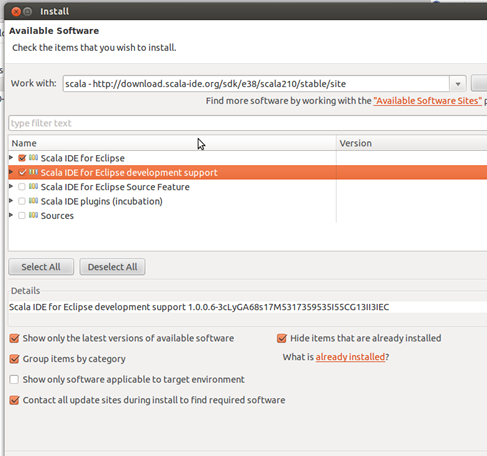
调试applications时，修改debug的前执行编译的命令为sbt的compile。

# Eclipse

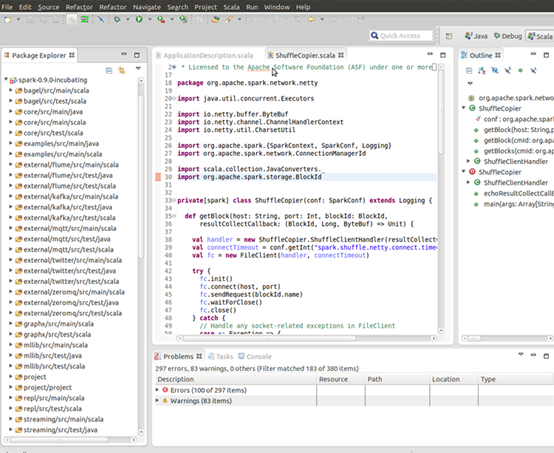
1. 安装Eclipse（<https://www.eclipse.org/downloads/>），下载解压后运行eclipse。
2. 在eclipse中安装Scala-ide plugin，在"Help" -> "Install New Software" -> "add" 中输入<http://download.scala-ide.org/sdk/e38/scala210/stable/site/>。



* 点击ok后，选择前两项，点击next后确认安装。



1. 安装成功后，就可以加载spark工程。通过"New"->"Scala Project" ->"Use default location"，选择已经编译后的Spark工程，确认后即可加载。加载后的Spark工程如下图所示。



# Spark编译

安装maven和scala

注意各自版本要求。

## maven

1. 直接使用mvn

注意设置JMV的参数， 避免OOM之类的错误。

比如设置：export MAVEN\_OPTS="-Xmx4096m -XX:MaxPermSize=4096m"

mvn -X -Pyarn -P[Hadoop](http://www.linuxidc.com/topicnews.aspx?tid=13)-2.2 -Dhadoop.version=2.2.0 -DskipTests clean package

1. 使用封装脚本

脚本中已经配置好JVM参数，直接使用即可。

另外：

* --tachyon的参数，不能放在前面，否则执行时不识别；
* 在配置好可用镜像之后，如果在download的过程中失败，可以多试几次 —— 网络因素导致；

## sbt

sbt编译过程比较慢。如果运行sbt时报sbt-launch.jar的错误，那么需要手动的安装，参考[这里](http://www.scala-sbt.org/0.13/tutorial/Manual-Installation.html)。

使用以下命令可以开始编译：

./sbt/sbt assembly

有时候由于网络问题可能导致编译hang在某个地方，或者你重启了，那么下次开始编译前需要删除$HOME/.ivy2/

.sbt.ivy.lock

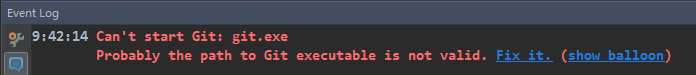
然后重启开始。sbt会进行增量编译。

# Git工具相关

## 官网 github 工具

### IDEA找不到 git.exe

打开IDEA，有如下错误提示：



1. 环境变量设置

在PATH中添加github安装目录的特定子目录下的bin目录到PATH中，如：安装目录\GitHub\PortableGit\_....\bin。设置后打开cmd命令行：



1. 可能遇到的问题

官网安装的 Github中 bin 所在目录会修改，如下：

C:\Users\lenovo\AppData\Local\GitHub\PortableGit\_ed44d00daa128db527396557813e7b68709ed0e2

C:\Users\lenovo\AppData\Local\GitHub\PortableGit\_c2ba306e536fdf878271f7fe636a147ff37326ad

导致IDEA找不到 git.exe

可能原因：在github界面工具上进行了用户切换操作导致 —— 未测试确认