

## Apache Zeppelin使用入门指南：安装

[Apache Zeppelin使用入门指南：安装](#)

[Apache Zeppelin使用入门指南：编程](#)

[Apache Zeppelin使用入门指南：添加外部依赖](#)

Apache Zeppelin是一款基于web的notebook(类似于ipython的notebook)，支持交互式地数据分析。原生就支持Spark、Scala、SQL、shell、markdown等。而且它是完全开源的，目前还处于Apache孵化阶段。本文所有的操作都是基于Apache Zeppelin 0.6.0-incubating-SNAPSHOT，spark 1.5.2的。

本文将介绍如何编译、安装以及使用Apache Zeppelin。

### 编译和安装Apache Zeppelin

Apache Zeppelin官方提供了Source包和二进制包，我们可以根据需要下载相关的包进行安装。这里我通过编译源码的方式来安装Apache Zeppelin，通过源码编译Zeppelin非常地简单，我这里从Zeppelin的git库里面下载最新的源码进行编译，步骤如下：

```
[iteblog@www.iteblog.com ~]$ git clone https://github.com/apache/incubator-zeppelin.git
[iteblog@www.iteblog.com ~]$ cd incubator-zeppelin
[iteblog@www.iteblog.com ~]$ mvn package -Pspark-1.5 -Dhadoop.version=2.2.0 -Phadoop-2.2 -DskipTests
[INFO] Reactor Summary:
[INFO]
[INFO] Zeppelin ..... SUCCESS [ 4.336 s]
[INFO] Zeppelin: Interpreter ..... SUCCESS [ 4.525 s]
[INFO] Zeppelin: Zengine ..... SUCCESS [ 2.465 s]
[INFO] Zeppelin: Spark dependencies ..... SUCCESS [ 30.643 s]
[INFO] Zeppelin: Spark ..... SUCCESS [ 2.220 s]
[INFO] Zeppelin: Markdown interpreter ..... SUCCESS [ 0.223 s]
[INFO] Zeppelin: Angular interpreter ..... SUCCESS [ 0.495 s]
[INFO] Zeppelin: Shell interpreter ..... SUCCESS [ 0.203 s]
[INFO] Zeppelin: Hive interpreter ..... SUCCESS [ 1.590 s]
[INFO] Zeppelin: Apache Phoenix Interpreter ..... SUCCESS [ 2.196 s]
[INFO] Zeppelin: PostgreSQL interpreter ..... SUCCESS [ 0.202 s]
[INFO] Zeppelin: JDBC interpreter ..... SUCCESS [ 0.171 s]
[INFO] Zeppelin: Tajo interpreter ..... SUCCESS [ 0.436 s]
[INFO] Zeppelin: Flink ..... SUCCESS [ 0.864 s]
[INFO] Zeppelin: Apache Ignite interpreter ..... SUCCESS [ 0.219 s]
[INFO] Zeppelin: Kylin interpreter ..... SUCCESS [ 0.223 s]
[INFO] Zeppelin: Lens interpreter ..... SUCCESS [ 1.030 s]
[INFO] Zeppelin: Cassandra ..... SUCCESS [ 2.954 s]
[INFO] Zeppelin: Elasticsearch interpreter ..... SUCCESS [ 1.601 s]
```

```
[INFO] Zeppelin: web Application ..... SUCCESS [01:19 min]
[INFO] Zeppelin: Server ..... SUCCESS [ 54.665 s]
[INFO] Zeppelin: Packaging distribution ..... SUCCESS [ 0.991 s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 03:12 min
[INFO] Finished at: 2016-01-20T16:35:33+08:00
[INFO] Final Memory: 97M/1298M
[INFO] -----
```

在编译Zeppelin的时候，会安装npm和node等，所以安装的进度有时会很慢。目前Zeppelin支持Spark 1.1到Spark 1.6和Hadoop 0.23到Hadoop 2.6等版本，我们可以分别通过下面选项指定相应的Spark和Hadoop版本：

```
-Pspark-1.6
-Pspark-1.5
-Pspark-1.4
-Pspark-1.3
-Pspark-1.2
-Pspark-1.1
-Pcassandra-spark-1.5
-Pcassandra-spark-1.4
-Pcassandra-spark-1.3
-Pcassandra-spark-1.2
-Pcassandra-spark-1.1

-Phadoop-0.23
-Phadoop-1
-Phadoop-2.2
-Phadoop-2.3
-Phadoop-2.4
-Phadoop-2.6
```

如果你需要使用到YARN，你必须在编译Zeppelin的时候指定-Pyarn选项，具体可以参见本博客的[《在Yarn上运行Apache Zeppelin & Spark》](#)。

如果你需要使用到PySpark，编译的时候需要使用-Ppyspark。

编译完Zeppelin之后，我们就可以启动Zeppelin：

```
[iteblog@www.iteblog.com ~]$ bin/zeppelin-daemon.sh start
```

停止Zeppelin可以使用：

```
[iteblog@www.iteblog.com ~]$ bin/zeppelin-daemon.sh stop
```

所有运行的日志会输入到logs目录下，如果遇到错误可以到里面去查找相关的日志。

需要注意的是，Zeppelin默认是在8080端口上启动相关的web服务的，在你服务器上，如果这个端口已经被占用了，那么会导致Zeppelin启动失败，并在日志里面抛出以下的异常：

```
WARN [2016-01-20 16:36:44,830] ({main} AbstractLifeCycle.java[setFailed]:204) -
FAILED SelectChannelConnector@0.0.0.0:8080: java.net.BindException: Address already in use
java.net.BindException: Address already in use
    at sun.nio.ch.Net.bind0(Native Method)
    at sun.nio.ch.Net.bind(Net.java:444)
    at sun.nio.ch.Net.bind(Net.java:436)
    at sun.nio.ch.ServerSocketChannelImpl.bind(ServerSocketChannelImpl.java:214)
    at sun.nio.ch.ServerSocketAdaptor.bind(ServerSocketAdaptor.java:74)
    at org.eclipse.jetty.server.nio.SelectChannelConnector.open(SelectChannelConnector.java:1
87)
    at org.eclipse.jetty.server.AbstractConnector.doStart(AbstractConnector.java:316)
    at org.eclipse.jetty.server.nio.SelectChannelConnector.doStart(SelectChannelConnector.java
:265)
    at org.eclipse.jetty.util.component.AbstractLifeCycle.start(AbstractLifeCycle.java:64)
    at org.eclipse.jetty.server.Server.doStart(Server.java:293)
    at org.eclipse.jetty.util.component.AbstractLifeCycle.start(AbstractLifeCycle.java:64)
    at org.apache.zeppelin.server.ZeppelinServer.main(ZeppelinServer.java:114)
```

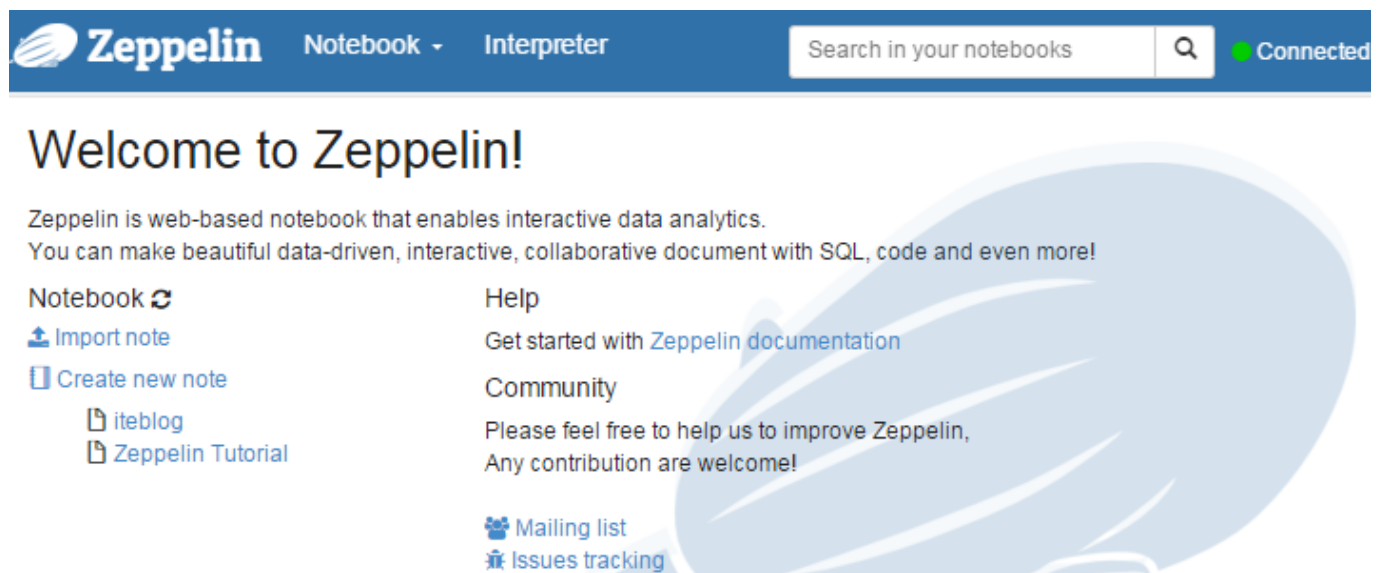
我们可以到conf/zeppelin-site.xml文件里面进行配置，并修改zeppelin.server.port属性的值为其他不被占用的端口，如下：

```
<property>
  <name>zeppelin.server.port</name>
  <value>8090</value>
  <description>Server port.</description>
</property>
```

然后我们需要重启Zeppelin：

```
[iteblog@www.iteblog.com ~]$ bin/zeppelin-daemon restart
```

一切顺利的话，我们就可以在<https://www.iteblog.com:8080>上面看到下面页面信息：



The screenshot shows the Apache Zeppelin web interface. The header is blue with the Zeppelin logo and the text 'Zeppelin Notebook - Interpreter'. There is a search bar on the right with the text 'Search in your notebooks' and a magnifying glass icon. A green dot indicates 'Connected'. The main content area has a large blue graphic and the text 'Welcome to Zeppelin!'. Below this, it says 'Zeppelin is web-based notebook that enables interactive data analytics. You can make beautiful data-driven, interactive, collaborative document with SQL, code and even more!'. There are two columns of links: 'Notebook' with 'Import note' and 'Create new note' (with sub-links 'iteblog' and 'Zeppelin Tutorial'), and 'Help' with 'Get started with Zeppelin documentation'. The 'Community' section includes 'Please feel free to help us to improve Zeppelin, Any contribution are welcome!' and links for 'Mailing list' and 'Issues tracking'.

本博客文章除特别声明，全部都是原创！  
转载本文请加上：转载自过往记忆（<https://www.iteblog.com/>）  
本文链接: 【】（）