

# FISCO BCOS零基础入门，五步轻松构建应用

原创 李辉忠 [FISCO BCOS开源社区](#) 2019-07-10



**李辉忠**

FISCO BCOS高级架构师

— AUTHOR | 作者 —

本文面向零区块链基础入门 FISCO BCOS的开发者，以高纯度、超浓缩的极简方式，分享如何快速基于 FISCO BCOS 构建你的第一个DAPP应用。

社区经常有人问：FISCO BCOS项目有10W+行源代码，10W+字说明文档，几十个子项目，我该如何下手，怎样入门？

莫慌，五步入门宝典已备好!!! 📌📌📌

## 第一步：构建一条FISCO BCOS的链

安装文档里给了一箩筐事无巨细的介绍，但此文就是要简明扼要地告诉你：**三步就能召唤 FISCO BCOS !**

(请先在home目录创建fisco目录，接下来都在这个目录操作)

```
1 $ curl -LO https://github.com/FISCO-BCOS/FISCO-BCOS/releases/download/  
2 $ bash build_chain.sh -l "127.0.0.1:4" -p 30300,20200,8545  
3 $ bash nodes/127.0.0.1/start_all.sh
```

你可以通过执行`ps -ef | grep -v grep | grep fisco-bcos`查看到四个节点已经在运行。

```
501 55515      1  0  3:00下午 ttys000      0:00.54 /Users/wheatli/fisco/nodes/127.0.0.1/node0/../../fisco-bcos -c config.ini
501 55517      1  0  3:00下午 ttys000      0:00.56 /Users/wheatli/fisco/nodes/127.0.0.1/node3/../../fisco-bcos -c config.ini
501 55518      1  0  3:00下午 ttys000      0:00.52 /Users/wheatli/fisco/nodes/127.0.0.1/node2/../../fisco-bcos -c config.ini
501 55520      1  0  3:00下午 ttys000      0:00.53 /Users/wheatli/fisco/nodes/127.0.0.1/node1/../../fisco-bcos -c config.ini
```

当然，如果你还是需要阅读详细文档，请参考【安装】

<https://fisco-bcos->

[documentation.readthedocs.io/zh\\_CN/latest/docs/installation.html#fisco-bcos](https://fisco-bcos-documentation.readthedocs.io/zh_CN/latest/docs/installation.html#fisco-bcos)

## 第二步：安装一个交互式控制台

控制台是一个可以交互式访问区块链，进行区块链数据读写请求的工具。无需太多解释，**四步完成控制台安装**：

```
1 $ bash <(curl -s https://raw.githubusercontent.com/FISCO-BCOS/console/
2 $ cp -n console/conf/applicationContext-sample.xml console/conf/applic
3 $ cp nodes/127.0.0.1/sdk/* console/conf/
4 $ cd console && ./start.sh
```

此时，你已经进入控制台界面，可以通过`help`查看命令列表，通过`getPeers`获取节点连接列表，通过`exit`或`quit`命令退出控制台。

同时，控制台内置了一个HelloWorld合约，可以直接调用`deploy HelloWorld`进行部署，然后调用`call HelloWorld`进行访问。



// 查询资产金额

```
function select(string account) public constant returns(int256, uint256)
```

// 资产注册

```
function register(string account, uint256 amount) public returns(int256)
```

// 资产转移

```
function transfer(string from_asset_account, string to_asset_account, uint256 amount)
public returns(int256)
```

- 逻辑实现：基于**CRUD**接口，实现业务逻辑

asset.sol合约：

<https://fisco-bcos->

[documentation.readthedocs.io/zh\\_CN/latest/docs/tutorial/sdk\\_application.html](https://fisco-bcos-documentation.readthedocs.io/zh_CN/latest/docs/tutorial/sdk_application.html)

## 第四步：合约编译与部署

Solidity合约需要通过编译器转换成机器（虚拟机）可执行的二进制，这些二进制是一系列OpCode的组合，虚拟机将解析执行这些OpCode实现合约业务逻辑。

编译之后的合约需要通过工具部署到区块链（写入区块链账本），之后可以根据合约接口描述文件（ABI）进行调用访问。

额，老毛病犯了，又唠唠叨叨讲原理，还是讲讲**如何无脑一键完成合约编译与部署**吧：

参考说明文档【控制台】的deploy命令

<https://fisco-bcos->

[documentation.readthedocs.io/zh\\_CN/latest/docs/manual/console.html](https://fisco-bcos-documentation.readthedocs.io/zh_CN/latest/docs/manual/console.html)

将Assert.sol放置在console/solidity/contract目录，在控制台执行deploy Assert.sol即可完成合约的编译以及部署。

[illegible]

## 第五步：开发业务

继续假设采用Java开发业务，当然也假设你熟悉eclipse、gradle、spring等常用工具。

- 1、建立一个Gradle Java工程asset-client，通过IntelliJ IDEA或者Eclipse；
- 2、编译build.gradle，增加maven库依赖；

```
repositories增加：maven {url
"http: //maven.aliyun.com/nexus/content/groups/public/"}maven{url
"https: //dl.bintray.com/ethereum/maven/" }
```

**dependencies**增加: compile ('org.fisco-bcos:web3sdk:2.0.4'){exclude group: 'org.ethereum'}

- 3、将第二步中控制台配置目录（console/conf/）的相关配置文件（applicationContext.xml, log.properties, ca.crt, node.crt, node.key）拷贝到asset-client项目的main/resource目录；
- 4、将第四步中一键编译生成的java文件（console/solidity/java/\*）拷贝到asset-client项目的main/java目录；
- 5、在main/java目录新建AssetClient类，Asset.java已经实现了有deploy, load, select, register, transfer接口，通过调用这些接口实现业务相关逻辑。

具体代码可参考示例工程：

<https://github.com/FISCO-BCOS/LargeFiles/raw/master/tools/asset-app.tar.gz>

到这里，你已经完成第一个基于FISCO BCOS的应用开发！如你对开发流程有疑问或优化建议，可以通过公众号进入技术交流群和我们一起探讨。🎉🎉🎉

..... **FISCO BCOS** .....

**FISCO BCOS的代码完全开源且免费**

**下载地址↓↓↓**

**<https://github.com/FISCO-BCOS/FISCO-BCOS>**



长按“二维码”关注