LAB4实验报告

【实验名称】Fabric 开发并部署链码

姓名 李泓民 学号 PB18071495

【实验目的及要求】

本次实验的目标是自己开发可以在 Fabric 上运行的链码，并且成功部署在 Fabric 网络上，并调用各个功能进行测试。

我选择了B档实现：

> 实现一个能够体现增删改查功能的链码，参考官方的例子即可，应用的业务场景不是考察的重点。部署并正确调用链码，截图。提交源码和实验报告。

【实验原理】

在Fabric中，根据提供的服务不同，可以把服务节点分为三类：CA、Orderer和Peer。

- CA：用于提供Fabric中组织成员的身份注册和证书颁发
- Orderer：排序节点，搜集交易并排序出块，广播给其他组织的主节点
- Peer：背书、验证和存储节点，链码安装的节点。

实验使用的Fabric版本为release-2.2，所有概念、架构以及命令文档，都可以在官方文档中搜索翻阅 一个企业级区块链平台 — hyperledger-fabricdocs master 文档

本次实验的目标是自己开发可以在Fabric上运行的链码，并且成功部署在Fabric网络上，并调用各个功能进行测试。

组织名Peer，通道名bcchannel，都与上一次组织不同。（这里因为是orderer出了一点点问题，根据助教的修改，改了orderer接口和相关的命令）

虽然这个组织Peer中有很多的peer节点，但每个节点安装各自的链码，由节点代表组织为各自安装的链码背书。因为该通道中只有一个组织Peer，所以能够搜集到足够的背书来支持调用链码的交易上链。

【实验平台】

使用vscode的ssh连接到服务器进行代码编辑，远程服务器为222.195.70.188。

【实验步骤】

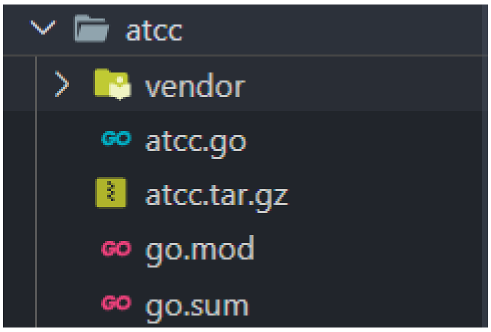
- 学习链码的开发，编写链码
  - 实现了银行账户存款的增删改查，每个账户提供唯一标识符id，并还有相关属性：owner，value
  - 在代码文件夹下面执行

    ```
    GO111MODULE=on go mod vendor
    ```

    下载依赖包存放于vender目录。

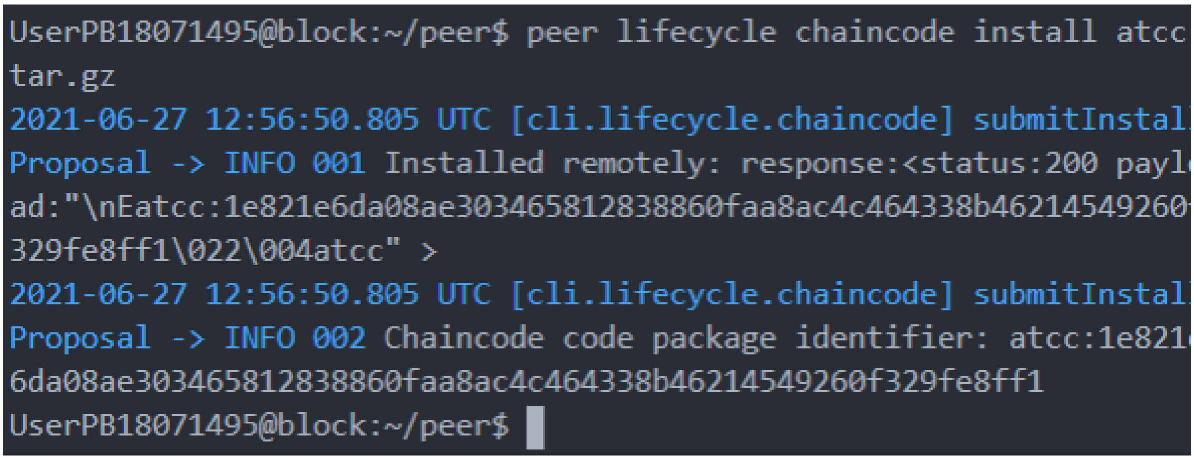
    然后在本文件夹之下压缩文件，打包链码 `peer lifecycle chaincode package atcc.tar.gz --path . --lang golang --label atcc`

文件夹结构如图：



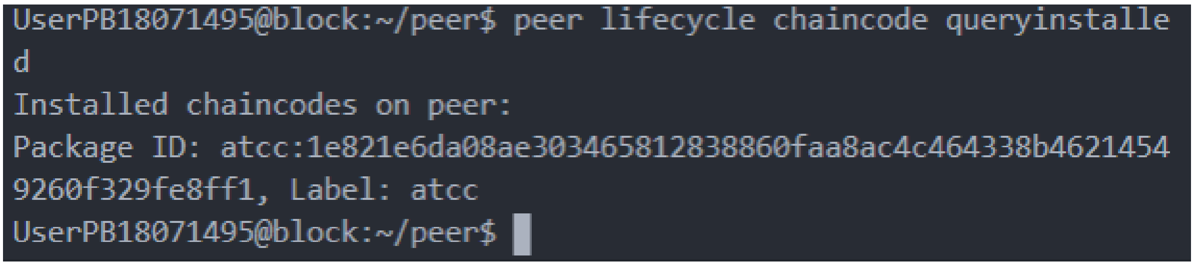
- 打包链码、安装链码、批准链码、实例化链码，在peer节点上完成

  安装链码：`peer lifecycle chaincode install atcc.tar.gz`



```
UserPB18071495@block:~/peer$ peer lifecycle chaincode install atcc
tar.gz
2021-06-27 12:56:50.805 UTC [cli.lifecycle.chaincode] submitInstal
Proposal -> INFO 001 Installed remotely: response:<status:200 payl
ad:"\nEatcc:1e821e6da08ae303465812838860faa8ac4c464338b46214549260
329fe8ff1\022\004atcc" >
2021-06-27 12:56:50.805 UTC [cli.lifecycle.chaincode] submitInstal
Proposal -> INFO 002 Chaincode code package identifier: atcc:1e821
6da08ae303465812838860faa8ac4c464338b46214549260f329fe8ff1
UserPB18071495@block:~/peer$ 
```

查询已经安装的链码信息：`peer lifecycle chaincode queryinstalled`

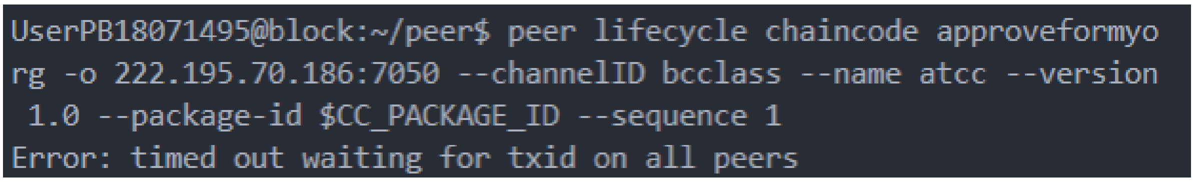


```
UserPB18071495@block:~/peer$ peer lifecycle chaincode queryinstalle
d
Installed chaincodes on peer:
Package ID: atcc:1e821e6da08ae303465812838860faa8ac4c464338b4621454
9260f329fe8ff1, Label: atcc
UserPB18071495@block:~/peer$ 
```

并且设置环境变量：

```
export
CC_PACKAGE_ID=atcc:1e821e6da08ae303465812838860faa8ac4c464338b46214549260f329
fe8ff1
```

在批准链码的时候遇到了一些问题，出现了如下错误：



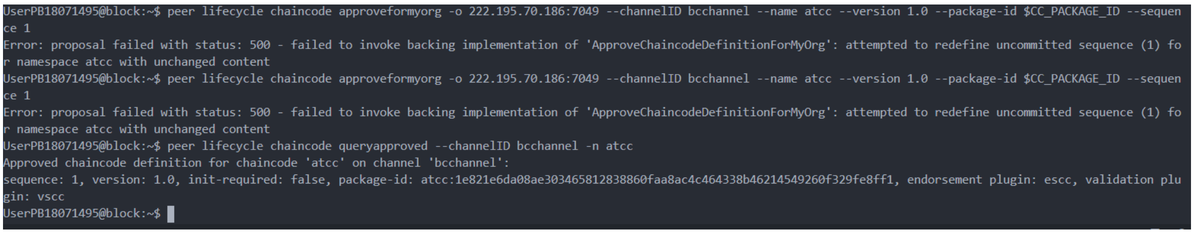
```
UserPB18071495@block:~/peer$ peer lifecycle chaincode approveformyo
rg -o 222.195.70.186:7050 --channelID bcclass --name atcc --version
 1.0 --package-id $CC_PACKAGE_ID --sequence 1
Error: timed out waiting for txid on all peers
```

后来通过修改orderer的端口重新注册peers解决，执行之后的效果如图：

```
peer lifecycle chaincode approveformyorg -o 222.195.70.186:7049 --channelID
bcchannel --name atcc --version 1.0 --package-id $CC_PACKAGE_ID --sequence 1
```

查询相关的批准的链码信息：

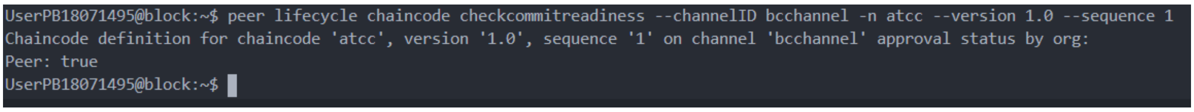
```
peer lifecycle chaincode queryapproved --channelID bcchannel -n atcc
```



```
UserPB18071495@block:~$ peer lifecycle chaincode approveformyorg -o 222.195.70.186:7049 --channelID bcchannel --name atcc --version 1.0 --package-id $CC_PACKAGE_ID --sequen
ce 1
Error: proposal failed with status: 500 - failed to invoke backing implementation of 'ApproveChaincodeDefinitionForMyOrg': attempted to redefine uncommitted sequence (1) fo
r namespace atcc with unchanged content
UserPB18071495@block:~$ peer lifecycle chaincode approveformyorg -o 222.195.70.186:7049 --channelID bcchannel --name atcc --version 1.0 --package-id $CC_PACKAGE_ID --sequen
ce 1
Error: proposal failed with status: 500 - failed to invoke backing implementation of 'ApproveChaincodeDefinitionForMyOrg': attempted to redefine uncommitted sequence (1) fo
r namespace atcc with unchanged content
UserPB18071495@block:~$ peer lifecycle chaincode queryapproved --channelID bcchannel -n atcc
Approved chaincode definition for chaincode 'atcc' on channel 'bcchannel':
sequence: 1, version: 1.0, init-required: false, package-id: atcc:1e821e6da08ae303465812838860faa8ac4c464338b46214549260f329fe8ff1, endorsement plugin: escc, validation plu
gin: vscc
UserPB18071495@block:~$
```

并且可以查询相关的已批准的组织信息：

```
peer lifecycle chaincode checkcommitreadiness --channelID bcchannel -n atcc --version 1.0 --sequence 1
```

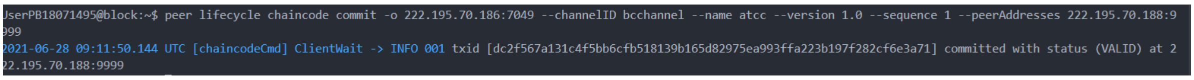


```
UserPB18071495@block:~$ peer lifecycle chaincode checkcommitreadiness --channelID bcchannel -n atcc --version 1.0 --sequence 1
Chaincode definition for chaincode 'atcc', version '1.0', sequence '1' on channel 'bcchannel' approval status by org:
Peer: true
UserPB18071495@block:~$
```

如果应该批准的大部分组织都已经批准了，那么就可以commit到通道（提醒排序节点和本peer），开始这个链码的服务：

```
peer lifecycle chaincode commit -o 222.195.70.186:7049 --channelID bcchannel --name atcc --version 1.0 --sequence 1 --peerAddresses 222.195.70.188:9999
```

因为该通道中只有一个组织Peer，所以能够搜集到足够的背书来支持调用链码的交易上链。



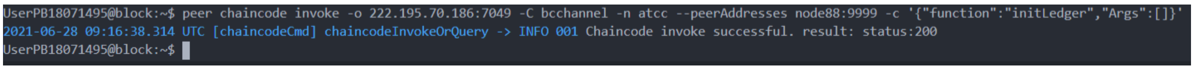
```
JserPB18071495@block:~$ peer lifecycle chaincode commit -o 222.195.70.186:7049 --channelID bcchannel --name atcc --version 1.0 --sequence 1 --peerAddresses 222.195.70.188:9
999
2021-06-28 09:11:50.144 UTC [chaincodeCmd] ClientWait -> INFO 001 txid [dc2f567a131c4f5bb6cfb518139b165d82975ea993ffa223b197f282cf6e3a71] committed with status (VALID) at 2
22.195.70.188:9999
```

【实验结果】

- 调用链码，查看功能是否实现

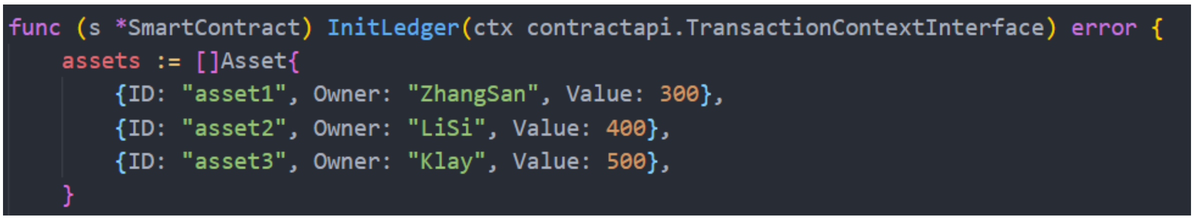
  最初的init，初始化数据库：

```
peer chaincode invoke -o 222.195.70.186:7049 -C bcchannel -n atcc --peerAddresses node88:9999 -c '{"function":"initLedger","Args":[]}'
```



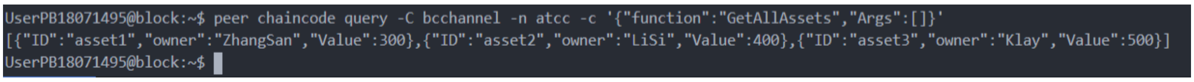
```
UserPB18071495@block:~$ peer chaincode invoke -o 222.195.70.186:7049 -C bcchannel -n atcc --peerAddresses node88:9999 -c '{"function":"initLedger","Args":[]}'
2021-06-28 09:16:38.314 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200
UserPB18071495@block:~$
```

在函数中对于初始化如下定义：



```
func (s *SmartContract) InitLedger(ctx contractapi.TransactionContextInterface) error {
    assets := []Asset{
        {ID: "asset1", Owner: "ZhangSan", Value: 300},
        {ID: "asset2", Owner: "LiSi", Value: 400},
        {ID: "asset3", Owner: "Klay", Value: 500},
    }
```

在终端输入后，显示如下：

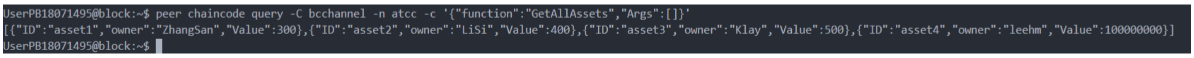


```
UserPB18071495@block:~$ peer chaincode query -C bcchannel -n atcc -c '{"function":"GetAllAssets","Args":[]}'
[{"ID":"asset1","owner":"ZhangSan","Value":300},{"ID":"asset2","owner":"LiSi","Value":400},{"ID":"asset3","owner":"Klay","Value":500}]
UserPB18071495@block:~$
```

可见成功的初始化了数据库。

再调用显示所有数据的 GetALLAssets 函数， 使用指令：

```
peer chaincode query -C bcchannel -n atcc -c
'{"function":"GetAllAssets","Args":[]}'
```

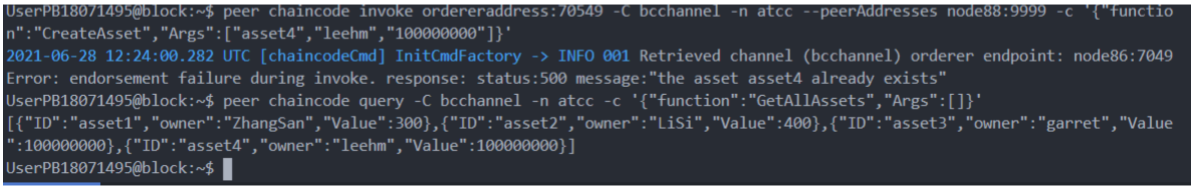


```
UserPB18071495@block:~$ peer chaincode query -C bcchannel -n atcc -c '{"function":"GetAllAssets","Args":[]}'
[{"ID":"asset1","owner":"ZhangSan","Value":300},{"ID":"asset2","owner":"LiSi","Value":400},{"ID":"asset3","owner":"Klay","Value":500},{"ID":"asset4","owner":"leehm","Value":100000000}]
UserPB18071495@block:~$
```

新建账户asset4：
执行指令：

```
peer chaincode invoke ordereraddress:7049 -C bcchannel -n atcc --
peerAddresses node88:9999 -c '{"function":"CreateAsset","Args":
["asset4","leehm","100000000"]}'
```

执行结果如下所示， 再调用 SearchCusInfo 函数查看是否真的添加成功：

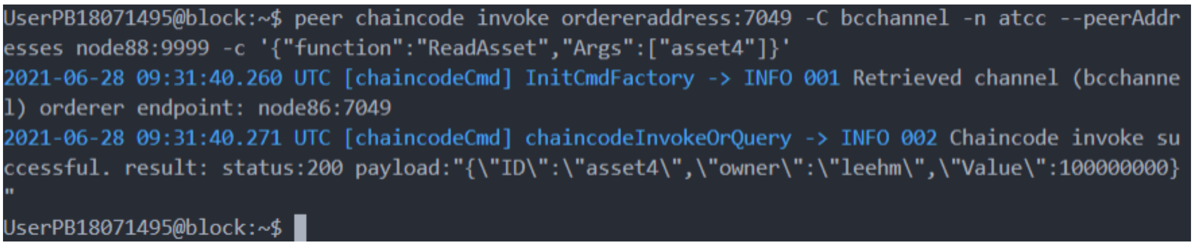


```
UserPB18071495@block:~$ peer chaincode invoke ordereraddress:70549 -C bcchannel -n atcc --peerAddresses node88:9999 -c '{"functio
n":"CreateAsset","Args":["asset4","leehm","100000000"]}'
2021-06-28 12:24:00.282 UTC [chaincodeCmd] InitCmdFactory -> INFO 001 Retrieved channel (bcchannel) orderer endpoint: node86:7049
Error: endorsement failure during invoke. response: status:500 message:"the asset asset4 already exists"
UserPB18071495@block:~$ peer chaincode query -C bcchannel -n atcc -c '{"function":"GetAllAssets","Args":[]}'
[{"ID":"asset1","owner":"ZhangSan","Value":300},{"ID":"asset2","owner":"LiSi","Value":400},{"ID":"asset3","owner":"garret","Value
":100000000},{"ID":"asset4","owner":"leehm","Value":100000000}]
UserPB18071495@block:~$
```

按 Id 值查找某一条记录， 例如查找asset4 ：
执行指令：

```
peer chaincode invoke ordereraddress:7049 -C bcchannel -n atcc --
peerAddresses node88:9999 -c '{"function":"ReadAsset","Args":["asset4"]}'
```

得到如下终端输出：

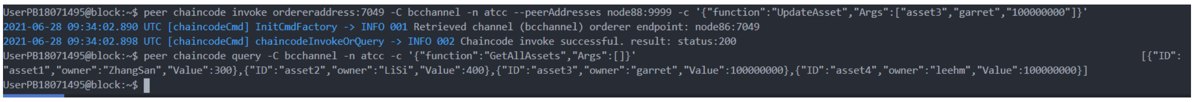


```
UserPB18071495@block:~$ peer chaincode invoke ordereraddress:7049 -C bcchannel -n atcc --peerAddr
esses node88:9999 -c '{"function":"ReadAsset","Args":["asset4"]}'
2021-06-28 09:31:40.260 UTC [chaincodeCmd] InitCmdFactory -> INFO 001 Retrieved channel (bcchanne
l) orderer endpoint: node86:7049
2021-06-28 09:31:40.271 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 002 Chaincode invoke su
ccessful. result: status:200 payload:"{\"ID\":\"asset4\",\"owner\":\"leehm\",\"Value\":100000000}
"
UserPB18071495@block:~$
```

修改记录： 以修改asset3的owner和value为例， 将其修改为 garrett，100000000
执行指令：

```
peer chaincode invoke ordereraddress:7049 -C bcchannel -n atcc --
peerAddresses node88:9999 -c '{"function":"UpdateAsset","Args":
["asset3","garret","100000000"]}'
```

得到结果如下所示， 再调用 GetAllAssets 函数查看是否真的修改成功：

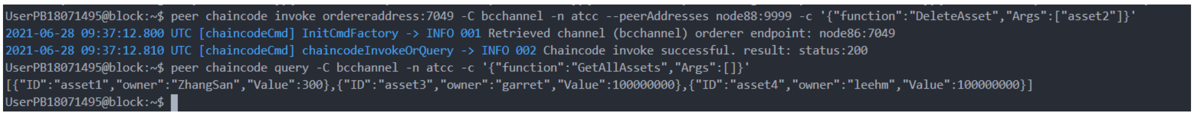


```
UserPB18071495@block:~$ peer chaincode invoke ordereraddress:7049 -C bcchannel -n atcc --peerAddresses node88:9999 -c '{"function":"UpdateAsset","Args":["asset3","garret","100000000"]}'
2021-06-28 09:34:02.890 UTC [chaincodeCmd] InitCmdFactory -> INFO 001 Retrieved channel (bcchannel) orderer endpoint: node86:7049
2021-06-28 09:34:02.898 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 002 Chaincode invoke successful. result: status:200
UserPB18071495@block:~$ peer chaincode query -C bcchannel -n atcc -c '{"function":"GetAllAssets","Args":[]}'                                        [{"ID":
"asset1","owner":"ZhangSan","Value":300},{"ID":"asset2","owner":"LiSi","Value":400},{"ID":"asset3","owner":"garret","Value":100000000},{"ID":"asset4","owner":"leehm","Value":100000000}]
UserPB18071495@block:~$
```

删除记录， 以删除asset2的记录为例：
执行指令：

```
peer chaincode invoke ordereraddress:7049 -C bcchannel -n atcc --
peerAddresses node88:9999 -c '{"function":"DeleteAsset","Args":["asset2"]}'
```

终端输出如下所示， 再调用 GetAllAssets 函数查看删除后的所有信息：



```
UserPB18071495@block:~$ peer chaincode invoke ordereraddress:7049 -C bcchannel -n atcc --peerAddresses node88:9999 -c '{"function":"DeleteAsset","Args":["asset2"]}'
2021-06-28 09:37:12.800 UTC [chaincodeCmd] InitCmdFactory -> INFO 001 Retrieved channel (bcchannel) orderer endpoint: node86:7049
2021-06-28 09:37:12.810 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 002 Chaincode invoke successful. result: status:200
UserPB18071495@block:~$ peer chaincode query -C bcchannel -n atcc -c '{"function":"GetAllAssets","Args":[]}'
[{"ID":"asset1","owner":"ZhangSan","Value":300},{"ID":"asset3","owner":"garret","Value":100000000},{"ID":"asset4","owner":"leehm","Value":100000000}]
UserPB18071495@block:~$
```

可见所有链码操作结果都符合预期。