腾讯云区块链服务平台介绍及接入

目录

1、TBaaS简介

1.1、TBaas简介

腾讯云区块链服务平台（TBaaS）是一个企业级的区块链开放平台，可一键式快速部署接入、拥有去中心化信任机制、集合众多区块链底层技术的区块链服务平台，目前已支持 Hyperledger Fabric 与 FISCO-BCOS 区块链底层平台，后续将支持 Corda、EEA 等不同区块链底层技术。平台目前支持私有链与联盟链两种模式，

TBaaS有一个重要特性是**多链支持**，这是其它普通联盟链没有的，不过从他的白皮书的了解到，TBaaS所说的多链并不是真正物理上的多链，而是TBaaS平台将自己的成员系统+通道组成的一个**具有用户权限的多Channel(通道)管理平台**，即逻辑上的多链，以下是官方白皮书对多链的说明。

> 一条逻辑上的区块链是集合了特定组织、特定节点的私有区块链系统，不同的组织间可以建立不同的逻辑区块链，链间实现数据隔离，智能合约可以部署在不同的逻辑区块链之上。
>
> 在TBaaS系统中，支持用户在同一个区块链系统中建立多个不同的逻辑区块链，即多链。多链 中每一条链都是包含记账节点、共识节点、智能合约和账簿的逻辑结构，它将参与者与数据(包含智能合约)进行隔离，实现了不同角色的用户访问权限不同，数据进行安全控制的基本要求。

[点击查看白皮书全文]

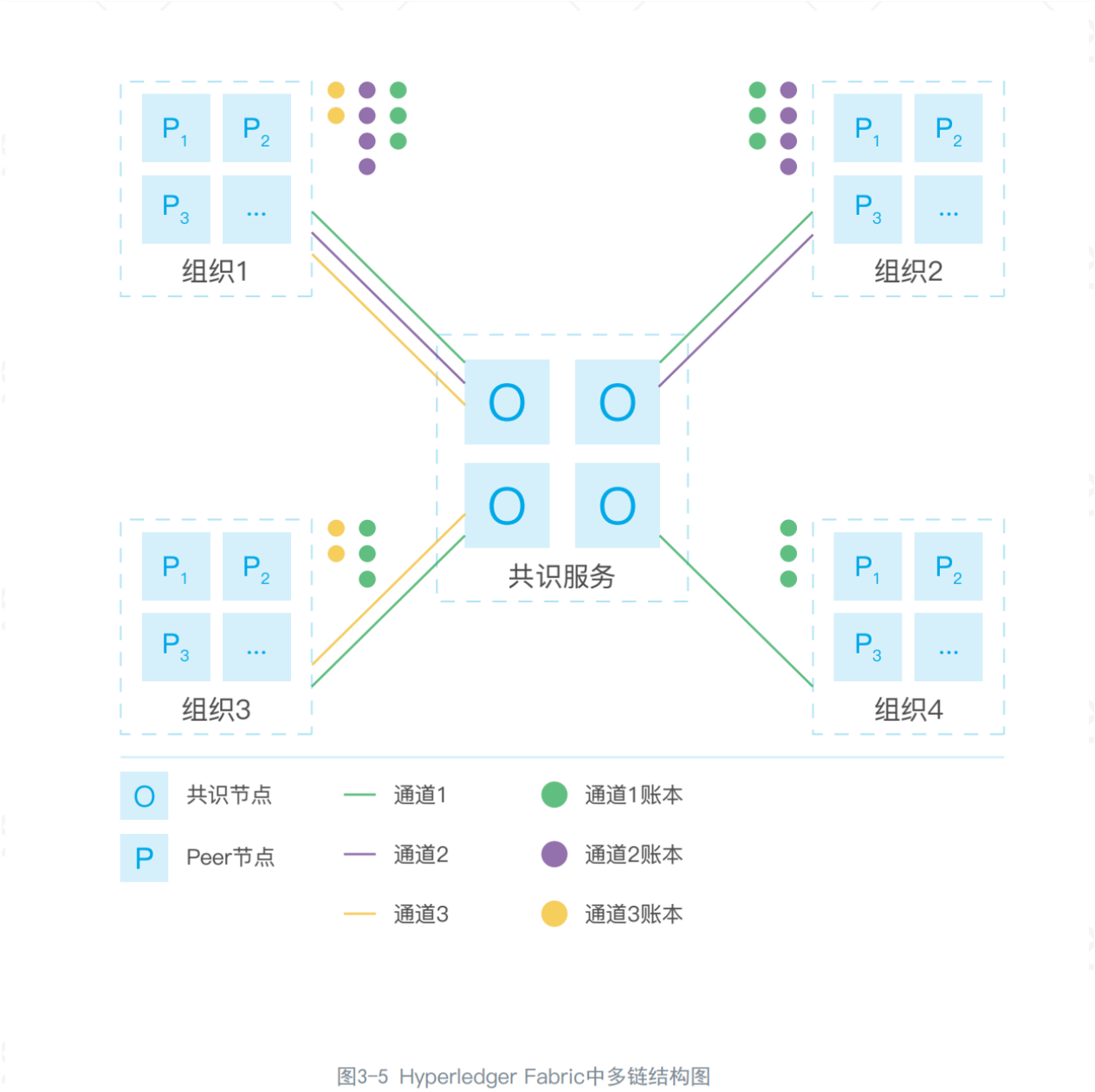


图3-5 Hyperledger Fabric中多链结构图

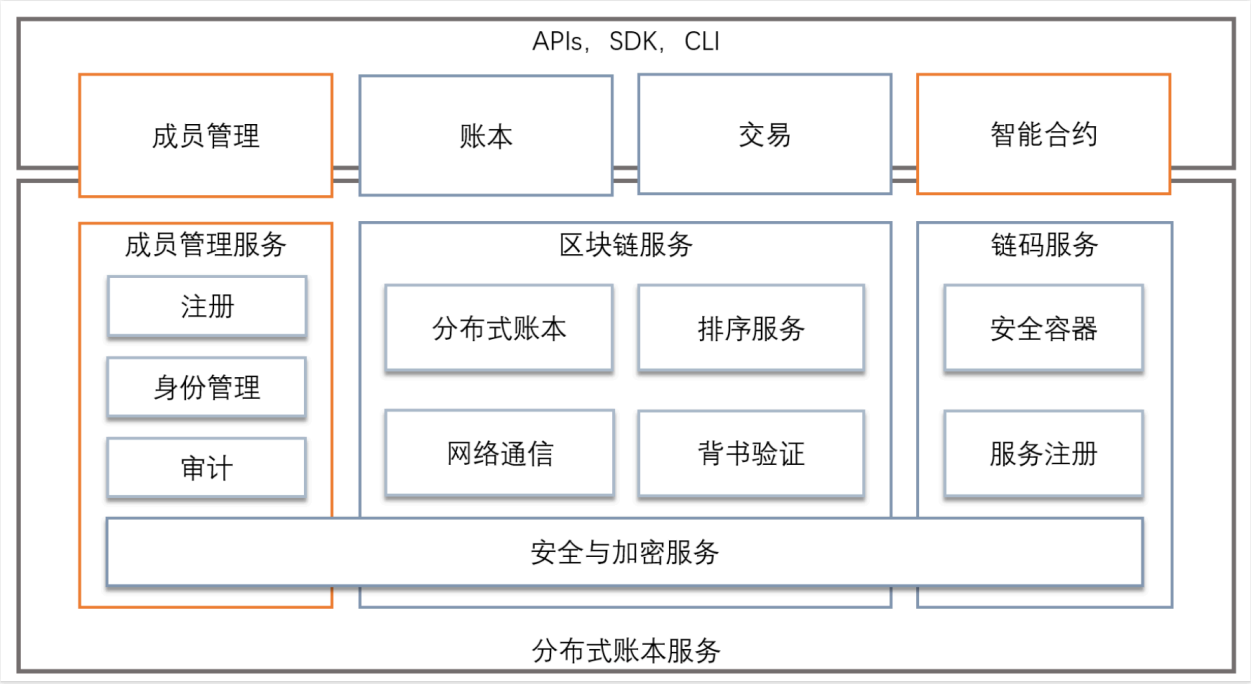
1.2、Hyperledger Fabric的架构

在说TBaaS之前，简单介绍一下Fabric。

Hyperledger Fabric 是一个企业级的区块链框架实现，是 Linux 基金会旗下托管的 Hyperledger 开源项目之一，具有高度的模块化、可配置设计，拥有完备的成员管理和治理措施，支持可插拔的共识协议。

Hyperledger Fabric 不需要利用传统的发币、挖矿、PoW 工作量证明等手段来激励参与方。针对联盟链场景，差异化设计使得 Fabric 成为当前性能优秀、广受认可的区块链底层引擎。

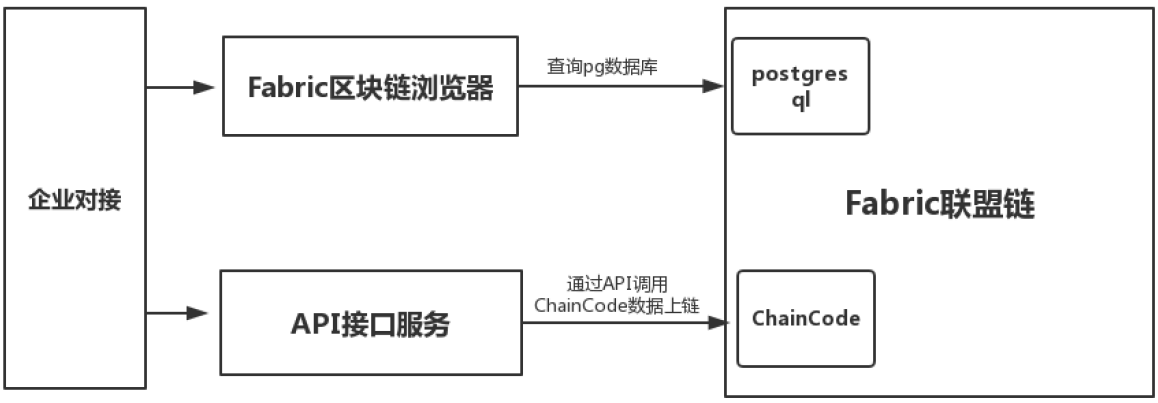
下面是Fabric整个架构图，



整个Fabric网络由成员管理服务、区块链服务、链码服务三大块组成，其中区块链服务就是整个架构中最重要、也是最难的部分，因为在整个部署过程中，docker服务很多，docker之前互相通信，只要其中一个服务启动有问题，就会影响整个联盟网络，所以操作起来非常复杂。

但在TBaaS中，Fabric作为TBaaS中底层引擎最重要的一种联盟链，区块链服务这块已由平台事先初始化好了，并且平台集成了腾讯云的用户系统，**TBaaS平台将Fabric的成员管理和腾讯云的用户系统打通**，这样在以后的成员管理，上链服务，多链上起到了很大的作用，同时作为企业，我们并不需要关心TBaaS底层的联盟链和用户管理的问题，我们只需要考虑ChainCode（智能合约）部分对接就可以。

如果一个项目需要将数据上传到Fabric联盟链中，整个Fabric项目基本会分为三大块，如下图



Fabric联盟链、Fabric区块链浏览器、API接口服务，Fabric联盟链是整个区块链的核心，联盟链中有什么，可以参考上面Fabric框架图，企业将数据上链，首先要编写Chaincode，将智能合约安装到通道中，再通过API接口服务调用ChainCode，将数据上链，企业还需搭建区块链浏览器，用于查询上链的数据，所以传统的Fabric联盟链中，企业需要做三块内容：

- 编写ChainCode
- 编写API接口服务
- 部署(编写)区块链浏览器

1.3、TBaaS的架构

TBaaS依托腾讯金融云基础设施，集开发、管理、运维和数据存储等功能为一体的一站式区块链服务平台。基于 TBaaS 区块链服务平台，客户可以降低对区块链底层技术的获取成本，专注在区块链业务模式创新及业务应用的开发和运营之中。TBaaS 区块链服务平台集合众多区块链底层技术，目前已支持 Hyperledger Fabric 与 FISCO-BCOS 区块链底层平台，后续将支持 Corda、EEA 等不同区块链底层技术，下面是TBaaS的整体架构图

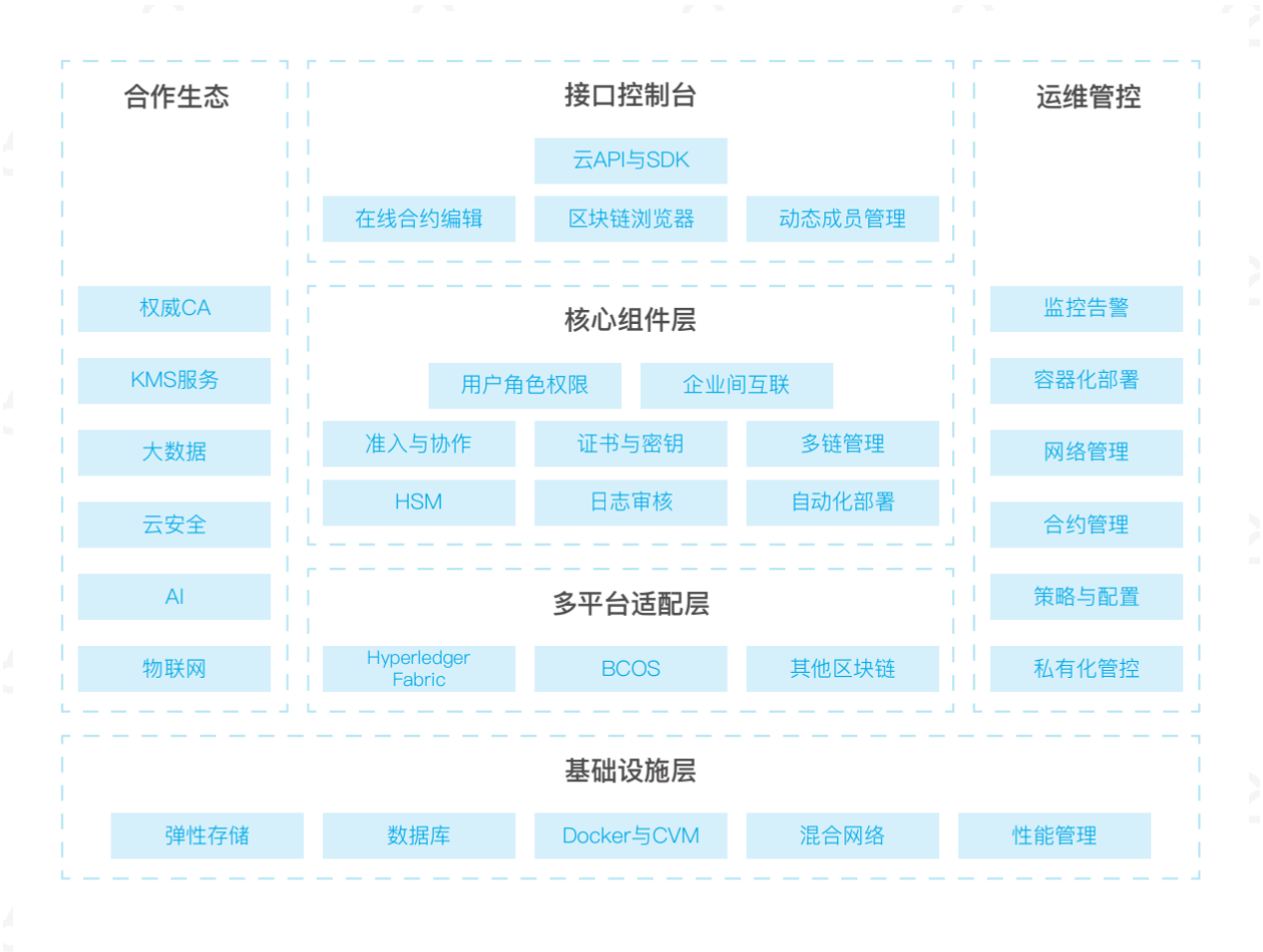


图3-3 腾讯云区块链总体的技术架构图

可以看到TBaaS架构比Fabric架构丰富了很多，TBaaS从创建联盟、成员管理、证书管理、合约的管理及在线编辑等都做了可视化界面，这大大缩减了开发人员的时间和成本，TBaaS还集成了运维管理，从监控警告到策略配置都有一套可视化界面，所以TBaaS把整个区块链服务做成了一个标准服务，这样企业可以更专注在区块链业务模式创新及业务应用的开发和运营之中。

2、如何使用TBaaS

2.1、创建联盟链

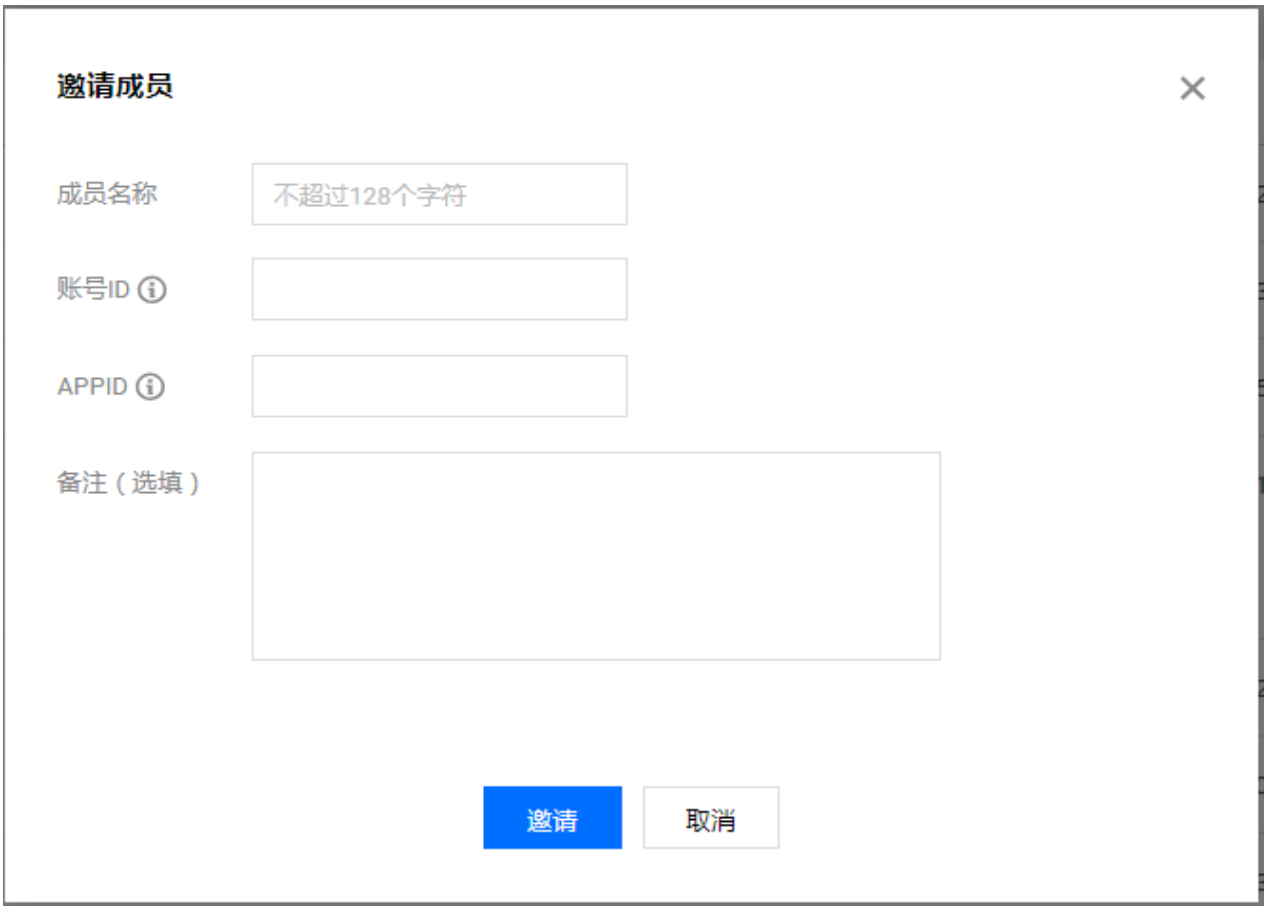
这里简要说明一下如何创建及管理整个TBaaS联盟链的过程，首先需要使用账号、密码登录腾讯云平台，进入管理后台通过 云产品->TBaas 进入 腾讯的TBaas管理后台。

2.1.1、创建联盟

联盟对应一个区块链业务团体，一个区块链业务团体可以由一个或者多个腾讯云用户组成。联盟可由腾讯云用户免费创建，在新建联盟过程中，只需要填写联盟名称、联盟的描述即可。

2.1.2、邀请成员

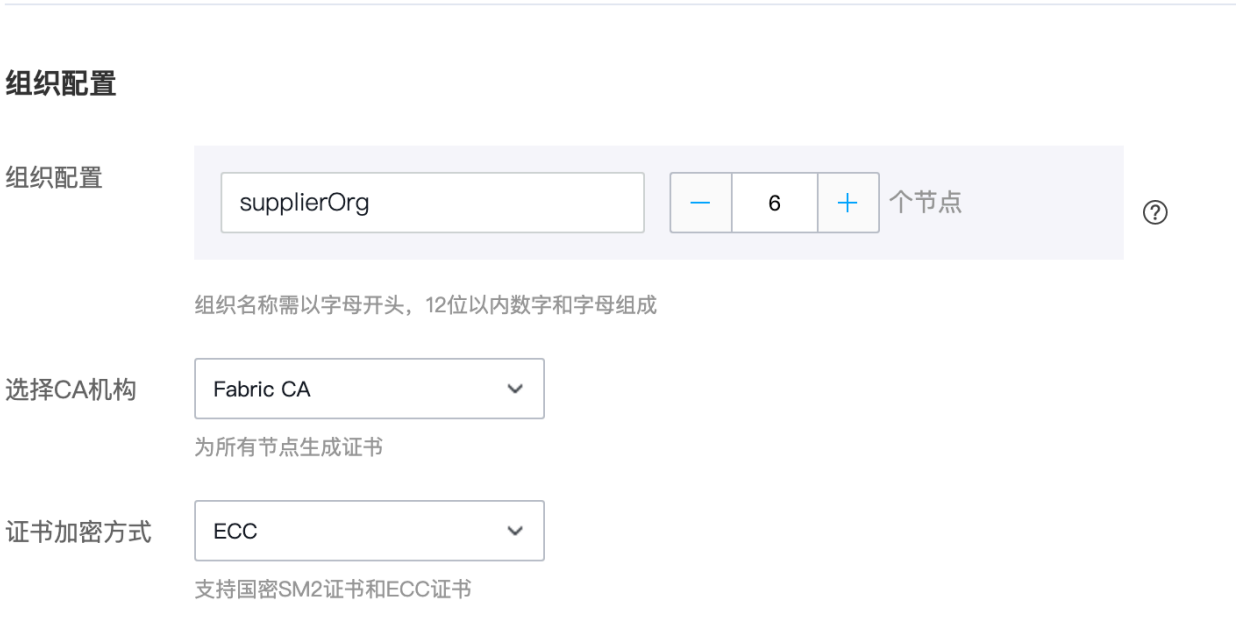
联盟成员可以邀请其他成员加入联盟，在 “联盟列表” 或者 “联盟详情” 页面中，单击 “邀请成员”，进行成员邀请。



2.1.3、创建区块链网络

创建联盟后，可以在该联盟下创建区块链网络及配置组织和节点

点击创建网络来创建一个联盟链网络



一个成员只能创建一个组织且每个组织最大只支持10个节点，

2.2、Channel管理

2.1的步骤走完后，整个联盟链的底层基础网络已经全部创建好，接下来需要创建通道，并将通道加入到组织中

2.2.1、创建Channel

在通道管理中点击新建智能合约。

2.2.2、 将Channel加入节点

在 “通道管理” 页签中，选择待加入节点的通道名称，单击【加入节点】。如下图所示：

2.2.3、邀请组织

一个联盟链中可以有多个通道，且每个通道可以邀请不同的组织加入该通道，

在 “通道管理” 页签中，选择待邀请组织的通道名称，单击【邀请组织】。如下图所示：

2.3、ChainCode(智能合约)管理

智能合约开发是区块链应用的主要功能，所有区块链业务能力围绕智能合约为核心，来实 现智能合同、自动触发、安全隔离、业务定义、数字协议等功能，因此智能合约是区块链应用开发 过程中最主要的部分，下面整个智能合约的生命周期图：

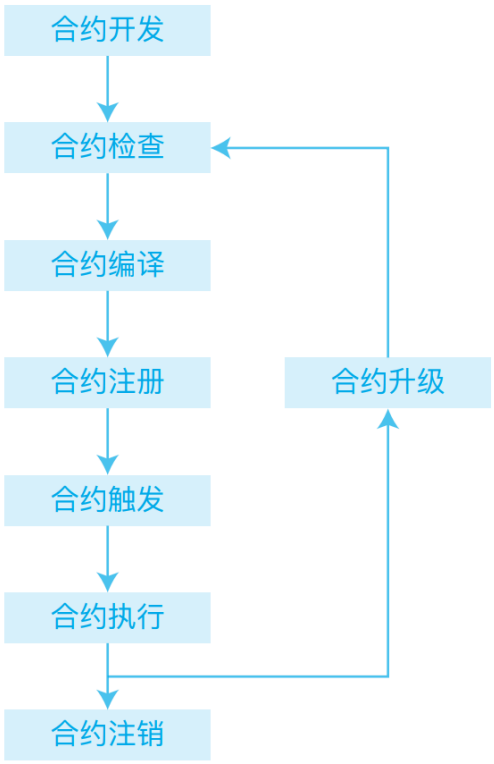


图3-6 智能合约服务

2.3.1、查看/新增合约

进入合约管理的 “区块链网络”，进入 “区块链网络” 信息页面，选择"合约管理"，可以在线新建一个合约，要注意的是，合约是用Go语言写的，如下图

单击【在线编辑】，可在区块链网络上编辑与使用智能合约。如下图所示： 具体操作可以按照使用指引使用智能合约 IDE 编辑器。

2.3.1 安装合约

点击下图合约列表右边的安装，可以将编写好的合约安装到某个组织里

2.3.2、实例化合约

安装好合约后，需要再实例化合约，点击**实例化合约，系统会启动一个Docker服务来运行这个智能合约**，如果在所有的组织里都安装、实例化智能合约，那会就会起相应数量的Docker服务来运行每个组织里的智能合约，注意**一定要先安装合约，再实例化合约**。

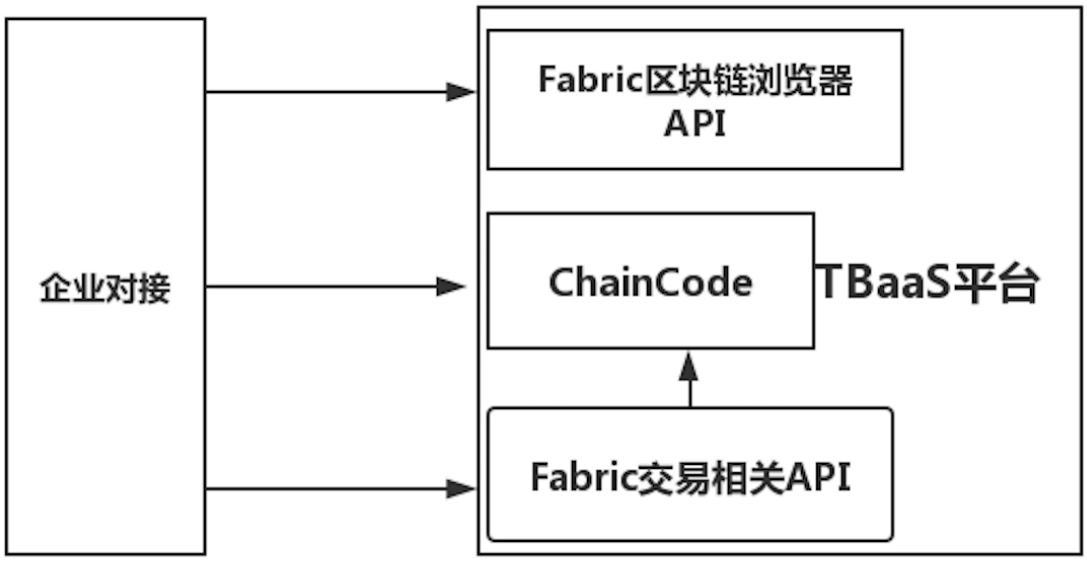
2.3.3、升级合约

有时候业务变动，需要修改合约内容，这时候修改后的合约内容要重新生效，就需要升级合约，同样如果在多个组织里都有对应的合约，则需要一个一个地升级。

3、数据如何上链

上面的步骤走完，整个TBaaS平台的区块链网络已经全部安装好，接下来就是要做企业如何对接TBaaS的联盟链，先来看下下面这张图，可以看到，TBaaS已经帮我们把整个网络都封装好了，我们只要调相关接口及编写智能合约就可以，整个项目对接只需要做如下三点即可：

- 编写ChainCode
- 调用TBaaS平台提供的Fabric交易相关API
- 调用TBaas平台提供的Fabric区块链浏览器相关API



3.1、编写ChainCode

TBaaS平台提供了一套很完善的在线编写智能合约的服务，使用在线IDE，可以直接编写和编译

需要注意的是，ChainCode官方文档显示可以使用Go,NodeJs,Java来编写合约，但是**官方推荐使用Go语言来编写智能合约**，合约具体内容根据我们业务来定，合约编写好后，具体的安装和实例化可以查看2.3的内容，

3.2、调用Fabric交易相关接口

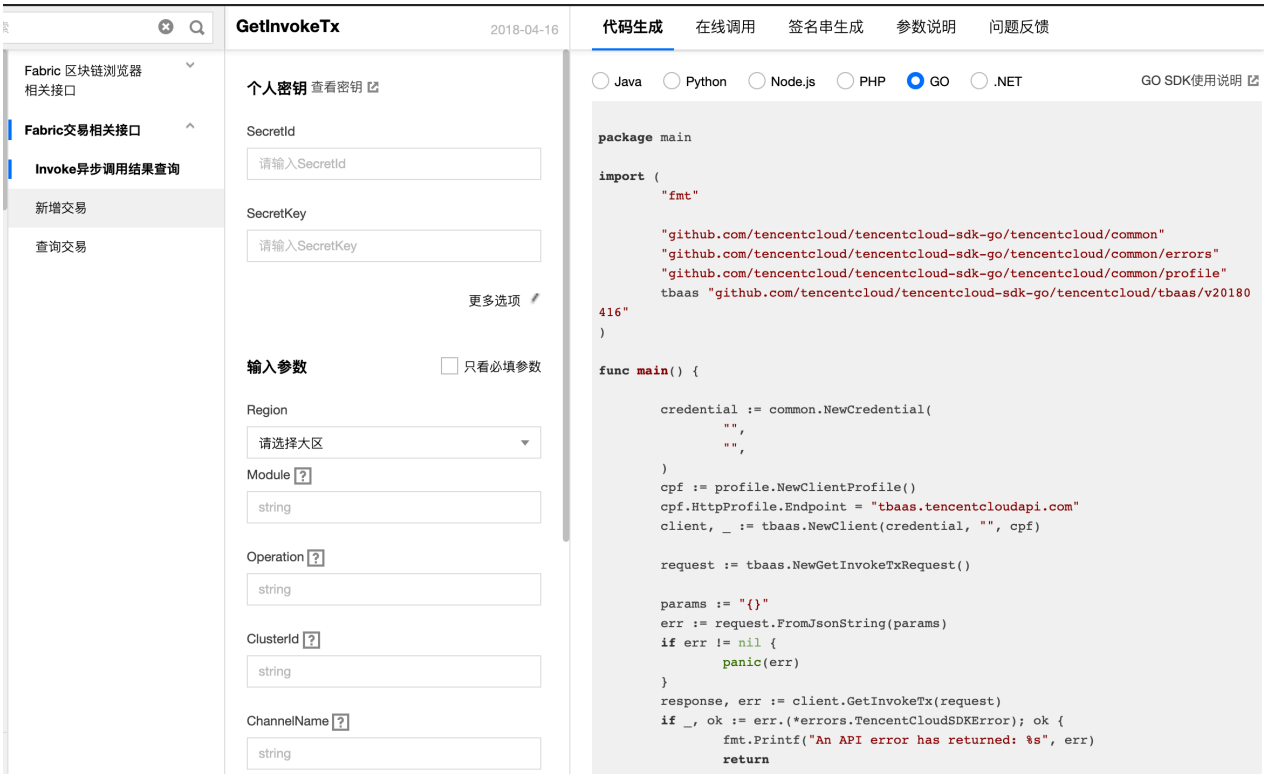
合约编写好后，通过安装和实例化后，就可以调用这个合约来将数据上链，TBaaS官方提供了三个Fabric交易相关的接口，分别是：

- Invoke异步调用结果查询
- 新增交易
- 交易查询

下面简要说下这三个接口的作用，详细文档可以查看官方文档

3.2.1、 Invoke异步调用结果查询

本接口是查询上链数据是否有效、成功，注意是异步的，TBaaS平台也提供了在线调试工具，如下图



注意调用Fabric相关的接口，官方提供了， Python、Java、NodeJs、PHP、GO语言的SDK，所以调用这些接口时，看我们自己技术方便，随便选用哪种语言的SDK都可以

下面是是 Invoke异步调用结果查询的一个官方示例，可以看下

输入示例

```
https://tbaas.tencentcloudapi.com/?Action=GetInvokeTx
&Module=transaction
&Operation=query_txid
&ClusterId=251005746envnew
&ChannelName=ch042103
&PeerName=peer0.neworg02.envnew
&PeerGroup=NewOrg02
&TxId=280e9f1436c3ce045af4f3c7060ff217583585d41faf1f1daa99387419bac07c
&GroupName=NewOrg02
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "BlockId": 6,
    "RequestId": "551b801e-6dbe-46be-aa46-f8cc3ff1cd09",
    "TxValidationCode": 0,
    "TxValidationMsg": "VALID"
  }
}
```

3.2.2、新增交易

这个接口应该是最重要的一个接口，就是将我们自己的数据上链，具体把什么数据上链，可以根据我们自己的业务系统来，数据上链成功后会返回一个Txid，这个交易ID是数据在链上的唯一标识，应该需要存储到我们的业务系统，以后可能用到这个Txid来查询链上的数据。

下面是一个示例：

输入示例

```
https://tbaas.tencentcloudapi.com/?Action=Invoke
&Module=transaction
&Operation=invoke
&ClusterId=251005746envnew
&Peers.0.PeerName=peer0.neworg02.envnew
&Peers.0.OrgName=NewOrg02
&ChannelName=ch042103
&ChaincodeName=cc050301
&FuncName=createCar
&Args.0=CAR92
&Args.1=Chevy
&Args.2=Volt
&Args.3=Black
&Args.4=Nick
&GroupName=NewOrg02
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "Events": "myOrgpeer0.myorg.envnew:VALID",
    "RequestId": "0b82b65e-7100-49f1-9f29-e934a8833711",
    "Txid": "0366ab8f31c9f8aa6b9fc9506fa841e55d1ecd492b3ecc373c0f66ca49f33ea1"
  }
}
```

3.2.3、交易查询

数据上链后，需要查看相关链上数据，注意这个查询接口是可以根据用户上链的数据查询，而不是根据上面的接口的Txid来查询，也就是说这个接口的查询交易的参数用户是可以自己自定义的，下面看下示例。

输入示例

```
https://tbaas.tencentcloudapi.com/?Action=Query
&Module=transaction
&Operation=query
&ClusterId=251005746envnew
&Peers.0.PeerName=peer0.neworg02.envnew
&Peers.0.OrgName=NewOrg02
&ChannelName=ch042103
&ChaincodeName=cc050301
&FuncName=queryCar
&Args.0=CAR92
&GroupName=NewOrg02
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "Data": [
      "
{\"make\":\"Chevy\",\"model\":\"Volt\",\"colour\":\"Black\",\"owner\":\"Nick\"
}"
    ],
    "RequestId": "3f6836c5-e889-431e-b932-47a1653c5f7b"
  }
}
```

可以看到Args.0=CAR92是自己定义的上链数据，并不是Txid。

3.3、调用Fabric区块链浏览器相关接口

TBaaS并没有直接提供给我们一个区块链浏览器相关的平台，只是提供了三个接口给我们，供我们自己查询，**我们可以通过这些接口自己做一个简单的TBaaS版区块链浏览器出来，实现区块链数据的可视化管理**，接口分别如下：

- 查询区块列表
- 获取区块链网络概要
- 获取最新交易列表

3.3.1、查询区块列表

这个接口可以查看链上所有的区块数据，分页展示，可以在我们自己的系统里把链上所有的数据都分页展示出来，可以看下示例：

输入示例

```
https://tbaas.tencentcloudapi.com/?Action=GetBlockList
&Module=block
&Operation=block_list
&ChannelId=0
&GroupId=0
&ChannelName=kylotst
&GroupName=liulanOrg
&ClusterId=251005746bc0f03q8u93j
&Offset=0
&Limit=10
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "BlockList": [
      {
        "BlockId": 5,
        "BlockNum": 5,
        "DataHash":
"92f1c3b1eb0eb1bc1825f1acd3474e7679b2a97cdf8d6155ed6a0e6c1458e479",
        "PreHash":
"92f1c3b1eb0eb1bc1825f1acd3474e7679b2a97cdf8d6155ed6a0e6c1458e479",
        "TxCount": 1
      },
      {
        "BlockId": 4,
        "BlockNum": 4,
        "DataHash":
"17e381e65605963c9211abc5b72d80cf3a6a4955ff7b61c70b406b98c90ded6f",
        "PreHash":
"17e381e65605963c9211abc5b72d80cf3a6a4955ff7b61c70b406b98c90ded6f",
        "TxCount": 1
      },
      {
        "BlockId": 3,
        "BlockNum": 3,
        "DataHash":
"32b80e93141467eda367f05a2428a6c03369405ae10349db7fad0762e9b56cc2",
        "PreHash":
"32b80e93141467eda367f05a2428a6c03369405ae10349db7fad0762e9b56cc2",
        "TxCount": 1
      },
      {
        "BlockId": 2,
```

```
        "BlockNum": 2,
        "DataHash":
"8be5094bf40c5b7e410cc3e8aa33354da2aa78db862e17cfd5fa783dddc1ee3a",
        "PreHash":
"8be5094bf40c5b7e410cc3e8aa33354da2aa78db862e17cfd5fa783dddc1ee3a",
        "TxCount": 1
      },
      {
        "BlockId": 1,
        "BlockNum": 1,
        "DataHash":
"af319fe2ff9e5830af0f07200adf965156446a205137e61d87f1db0b8f43cc7e",
        "PreHash":
"af319fe2ff9e5830af0f07200adf965156446a205137e61d87f1db0b8f43cc7e",
        "TxCount": 1
      },
      {
        "BlockId": 0,
        "BlockNum": 0,
        "DataHash":
"96f22d00947478c1b16c2aec5e7079c761b168d304cadf19a14a032ee8f64a23",
        "PreHash":
"96f22d00947478c1b16c2aec5e7079c761b168d304cadf19a14a032ee8f64a23",
        "TxCount": 1
      }
    ],
    "RequestId": "018328c8-9c24-4104-a0ad-a7a31c033278",
    "TotalCount": 6
  }
}
```


3.3.2、获取区块链网络概要

这个接口可以查看区块网络的一个概要信息，包括当前通道数量、组织数量、节点数量、合约数量、Orderer数量等，下面是一个示例：

输入示例

```
https://tbaas.tencentcloudapi.com/?Action=GetClusterSummary
&Module=cluster_mng
&Operation=cluster_summary
&ClusterId=251005746bc0f03q8u93j
&GroupId=0
&GroupName=liulanOrg
&<公共请求参数>
```

输出示例

```
{
  "Response": {
    "ClientCertCount": 2,
    "JoinChannelCount": 1,
    "MyChaincodeCount": 1,
    "MyChannelCount": 1,
    "MyGroupCount": 1,
    "MyPeerCount": 2,
    "NoneChannelCount": 0,
    "OrderCertCount": 0,
    "OtherChaincodeCount": 0,
    "OtherChannelCount": 0,
    "OtherGroupCount": 0,
    "OtherPeerCount": 0,
    "PeerCertCount": 2,
    "RecentChaincodeCount": 1,
    "RequestId": "8646a1d8-bae3-4b41-8732-06b8c004eaa5",
    "TlsCertCount": 4,
    "TotalCertCount": 8,
    "TotalChaincodeCount": 1,
    "TotalChannelCount": 1,
    "TotalGroupCount": 1,
    "TotalPeerCount": 2
  }
}
```


3.3.3、获取最新交易列表

这个接口也是获取交易数据，只是他获取的是最新上链的数据列表，总体跟3.3.1的接口很相似，下面是示例

输入示例

```
https://tbaas.tencentcloudapi.com/?Action=GetLatesdTransactionList
&Module=transaction
&Operation=latest_transaction_list
&GroupId=0&ChannelId=0
&LatestBlockNumber=5
&GroupName=liulanOrg
&ChannelName=kylotst
&ClusterId=251005746bc0f03q8u93j
&Offset=0
&Limit=10
&<公共请求参数>
```

输出示例


```
{
  "Response": {
```

```
    "RequestId": "68cd2009-6a2b-481e-850e-08522a546221",
    "TotalCount": 5,
    "TransactionList": [
      {
        "BlockHeight": 6,
        "BlockId": 5,
        "CreateOrgName": "liulanOrg",
        "CreateTime": "2019-04-24 11:39:52",
        "TransactionHash":
"da6a44da08bda02fb94b2b3fb684350a7f636044f348dbed9804f1dc143d2a01",
        "TransactionId":
"4ea1f77d1c2622f6672e37c611a0542dc9e21b15298b03de4127df454a17a457",
        "TransactionStatus": "VALID",
        "TransactionType": "ENDORSER_TRANSACTION"
      },
      {
        "BlockHeight": 5,
        "BlockId": 4,
        "CreateOrgName": "liulanOrg",
        "CreateTime": "2019-04-24 11:39:25",
        "TransactionHash":
"6e8a88088e1c0b111f7ff0f74e7891a15c5c32cd1fa68d8e5f8d11637fd4ca04",
        "TransactionId":
"9f2e40a4443d9928ed9b8266e893ac90ed00381227994281833ecd4393d40b15",
        "TransactionStatus": "VALID",
        "TransactionType": "ENDORSER_TRANSACTION"
      },
      {
        "BlockHeight": 4,
        "BlockId": 3,
        "CreateOrgName": "liulanOrg",
        "CreateTime": "2019-04-23 19:15:59",
        "TransactionHash":
"421234c8ad48052f202a262f1fe739b963831c423b2d0028ba7496eca837cac9",
        "TransactionId":
"bf2cdfd82a7b9a9a5ce135ef41687f3f04496c41575e2994fae99b58bec80754",
        "TransactionStatus": "VALID",
        "TransactionType": "ENDORSER_TRANSACTION"
      },
      {
        "BlockHeight": 3,
        "BlockId": 2,
        "CreateOrgName": "liulanOrg",
        "CreateTime": "2019-04-23 19:14:23",
        "TransactionHash":
"57407887234a4645c7b339aea5e94d3cf5f017f5eb05037a2c12fecf9d4fd6ae",
        "TransactionId":
"71594843087d3435c2aa74f69d9d995ed97fff4951778b86956c2c069691fb54",
        "TransactionStatus": "VALID",
```

```
        "TransactionType": "ENDORSER_TRANSACTION"
      },
      {
        "BlockHeight": 2,
        "BlockId": 1,
        "CreateOrgName": "liulanOrg",
        "CreateTime": "2019-04-23 14:34:29",
        "TransactionHash":
"9d798fe9f0eeaf5193ec1f9dc21e930bf3232820ddc1db18625b9f02bd35f54a",
        "TransactionId":
"4ee11735f87b34673ea88c162e71b20fb71f798e89231e45bc0fbc6b9f09d02c",
        "TransactionStatus": "VALID",
        "TransactionType": "ENDORSER_TRANSACTION"
      }
    ]
  }
}
```


4、需要注意的问题

上面是整个TBaaS的使用和对接过程，可以说基于TBaaS平台，联盟链的开发成本降了很多，但在整个开发过程中，有以下几个问题需要注意。

4.1、哪些数据适合存储在链上?

参与方最好不要将一些敏感信息上链，例如用户身份信息、用户联系方式、资金信息等。适合存储在链上的数据可划分为以下方面：

- 项目由多方参与，且数据需要多方达成共识才能生成。
- 项目由多方参与，且数据需要在多方之间内共享。
- 需要特定参与方对真实性进行背书的数据。
- 有直接或者间接价值的数据。

4.2、上链的性能问题

从TBaaS的官方文档上看，15节点的区块链网络中实现单通道超过7000TPS，官方并没有说明具体服务器的配置，之前我们做过测试，Farbic网络中，8G内存，4核心CPU，4个节点，单通道请求一般是200-300TPS左右，在实际应用中，一开始我们的节点数没有这么多，还有服务器的具体配置与应用环境，在对接的过程需要注意请求并发量，防止上链的性能问题。

4.3、上链的数据隐私问题

目前TBaaS针对Fabric的数据隐私问题，有如下不同程序保护

- 使用多链支持，将区块链分为若干通道+腾讯云用户系统，通道与通道之间的数据隔离存储和传输，数据只能被此通道的参与方访问。

- 参与方可以仅将数据的摘要值或者加密后的密文上链。该方式需要保证数据的访问方能够根据摘要值获取到原始数据，或者拥有对密文进行解密的密钥。
- 通过在智能合约中定义规则，只允许特定的角色有权限访问数据。

徐耀

最后更新时间：2019-06-06