# 区块链服务网络帮助手册

（Blockchain-based Service Network Help Manual）

Version 1.0.5

区块链服务网络发展联盟

# 目录

# 1 区块链服务网络介绍

## 1.1 简介

服务网络的设计和建设理念完全借鉴互联网:互联网是由TCP/IP协议将所有数据中心连接而形成的,服务网络是通过建立一套区块链运行环境协议将所有数据中心连接而组成。与互联网一样，服务网络也是跨云服务、跨门户、跨底层框架的全球性基础网络。

服务网络的直接参与方有三类：一是云服务商，通过安装免费的服务网络公共城市节点软件，将其云服务资源（CPU、存储和带宽）接入服务网络,并在服务网络上进行销售;二是区块链底层框架商(特指联盟链)，根据服务网络底层框架适配标准将框架进行适配后，可以部署到服务网络，供开发者选择使用；三是门户商，可以在已有的云服务门户或开发者门户内，通过服务网络快速并低成本地建立BaaS(Blockchain as a Service)平台，并向自己的客户提供基于服务网络的区块链应用开发、部署和运行服务。服务网络是一个开放性网络，任何云服务商、底层框架商和门户商在符合服务网络协议标准的前提下，均可以自由选择加入或退出服务网络。

与互联网一样，服务网络的直接使用者是开发者和科技公司。区块链应用开发者可以通过任意一个服务网络门户，在全世界任何公共城市节点上购买以TPS（Transactions Per Second）、存储量和带宽作为计费标准的云资源，并选择任何已适配的底层框架，以极低的成本和极方便的操作进行区块链应用的开发、部署和运营。区块链应用

的发布者只需将应用的所有记账节点部署到服务网络的一个或多个公共城市节点上，参与者即可通过任何一个公共城市节点网关进行几乎无成本的接入。在每个公共城市节点内，所有部署的应用共享服务器资源。对于高频应用，公共城市节点可以为其智能化地自动分配单独的高处理性能的记账节点；而对于低频应用，则可多个应用共享一个记账节点。这种资源共享的机制能使服务网络所提供的资源成本降低至传统区块链云服务所需成本的三分之一到五分之一。

服务网络是一个信息化基础设施，如同家家户户无需自己打井吃水，而是通过在城市建立公共水厂享受统一的供水服务，从而降低社会成本。在服务网络上，区块链应用发布者和参与者均不需要再购买物理服务器或者云服务来搭建自己的区块链运行环境，而是使用服务网络提供统一的公共服务，并按需租用共享资源，从而大大降低发布者和参与者的成本。经调研，如需搭建一个传统联盟链局域网环境，根据目前中国主流云服务商的报价，每年最低成本也要人民币十万元左右。而通过服务网络，一个应用每年仅需人民币 2-3 千元即可成链并进入运行。这样将鼓励大量的中小微企业、甚至学生在内的个人通过服务网络进行创新、创业，从而促进区块链技术的快速发展和普及。

总体而言，从传统区块链的孤立封闭架构发展到服务网络的资源共享架构，完全遵循了互联网从早期的众多封闭孤立局域网逐步扩展成为全球性互联互通设施的发展历程。服务网络可以被视为区块链互联网。

因为服务网络是多门户的，本帮助手册仅适用于 bsnbase.com，

不适用于其他 BSN 门户。

## 1.2 优势

- **提供公共城市节点：**区块链服务网络在每个城市建立联盟链公共城市节点，所有应用统一部署在一个或多个公共城市节点中，用户连接公共城市节点网关进行业务处理，不需要每个企业针对每个联盟链应用自建记账节点和搭建运行环境。公城市节点由服务网络统一运维，开发者只需专注于核心业务创新，不需花费精力在区块链环境的建设和维护上。
- **支持联盟链多底层框架：**区块链底层框架可以被视为是区块链应用的操作系统。目前，服务网络已经支持 Hyperledger Fabric，正在适配的联盟链底层框架有 Fabric 国密、FISCO BCOS、CITA、XuperChain、梧桐链和 Brochain 等。针对公链框架，目前支持以太坊和 EOS 等（无法在中国公共城市节点和门户内部署）。
- **负载均衡服务：**城市节点上的记账节点均具备灵活的负载均衡机制。当某应用需要更多计算力和 TPS 时，服务网络会自动为其分配更多的记账节点和共识排序资源。
- **智能网关和预制链码：**为了进一步降低开发者的成本，服务网络每个城市节点均配置了智能网关，可以由外部系统接入服务网络。服务网络门户提供基础预制链码包，使开发者在完全不懂区块链开发语言的情况下，使用自己熟悉的编程语言，即可如同连入传统数据库一般使用区块链服务网络。

- **应用部署和接入方便：**服务发布者发布区块链服务时，选择需要的城市节点、每个节点的 TPS、带宽、存储容量后，一键即可完成所有节点的应用部署。应用使用方可以通过互联网或专线随时连入就近城市节点使用应用。
- **实现跨链数据互通：**通过区块链服务网络的证书和权限机制，两个基于不同底层框架的区块链应用之间只要互相授权，即可实现数据的互相调用。
- **降低运营成本：**由应用服务发布方统一选择节点、带宽、TPS、存储量配置，并承担应用在服务网络上的费用。平均总价格只是区块链云服务的五分之一到三分之一。而应用服务的使用者可以免费连入服务网络使用相关应用。

## 1.3 名词解释

- **城市节点：**是服务网络的基础网络运行单元，部署在每个地级市的 IDC 机房或云资源内。城市节点在服务网络中承担数据信用责任，是服务网络的基础网络参与方，也是服务网络中的基础网络节点组织成员。多个城市节点即构成以区块链技术为基础的服务网络的基础网络联盟。
- **行业应用：**服务网络上运行的应用服务和在门户应用商店中应用产品的总称；
- **应用服务：**指在区块链服务网络中已经发布并运行的区块链应用，用户可以通过服务网络门户或发布方自己的业务系统直接申请参

与使用。

- **应用产品：**指没有在区块链服务网络中发布和运行，仅在服务网络门户的应用商店内上架的区块链应用。产品销售后，由购买方在服务网络上发布后方可以投入使用。产品发布完成之后即成为服务，此时购买者成为应用发布者，并可以邀请其他用户参与。产品的设置是让某些开发者仅以开发和销售产品为商业模式，而不需要投入成本去运营自己的产品（对比互联网，产品开发者类似提供网站搭建的开发者，而服务发布者类似经营网站的运营者）。
- **服务发布者：**发布区块链应用服务，拥有该应用产权，可以进行服务的运营和商务拓展管理，对通过门户网站的服务参与者的加入进行审批和操作权限管理。服务可以为公开服务和非公开服务。非公开服务将在门户网站内无法浏览和搜索到，由服务发布者内部使用或者通过自己业务系统管理使用者。
- **服务参与者：**是使用区块链服务的客户，可以使用发布方的业务系统或者使用自有业务系统接入相关应用使用其服务。
- **产品开发者：**是在服务网络门户的应用商店内发布产品信息的个人或企业开发者；
- **链下业务系统：**是区块链服务发布者或服务参与者部署在区块链服务网络之外的业务系统；链下业务系统通过区块链服务网络提供的开放接口向区块链服务网络获取或写入数据；

## 1.4 典型应用场景

区块链技术具有分布式存储、信息可追溯、数据不可篡改、数据安全等特点。区块链服务网络以区块链技术为基础，在全国范围内建立了数百个城市物理服务器节点。基于区块链技术的特点和区块链服务网络优势，可有效解决行业业务痛点，帮助企业实现业务模式创新。以下列举的是区块链服务网络的几个典型应用场景：

**场景一：商品溯源**

在传统的商业零售场景中，商品从出厂到消费者的链条存在中间环节信息不透明的情况。当发生商品质量或安全问题时，难以回溯和追责。同时供应链信息也存在被篡改或伪造的风险。

区块链在其中起到的作用是，可以实现多方共同记录溯源信息，让消费者可对商品实现溯源查询和验真。同时，区块链保证了溯源信息是经过各个业务参与方共同确认的，具备不可篡改性。区块链上的全流程交易历史记录均可被审计，满足政策和监管的要求。此外，区块链需要结合适用于各种商品形态的防伪以及数字化技术，才能形成一套完善的溯源方案。

**场景二：公益慈善**

基于区块链数据可追溯、不可篡改等特点对捐赠资金及资金流向进行公示，可以提高受捐机构的公信力，减少捐赠人对资金使用的质疑。

**场景三：供应链金融**

在传统供应链体系中，除了核心企业外，还包含了上游的多级供应商和下游的多级经销商，但由于企业的信用无法实现传递，导致许多企业面临融资难、融资慢的问题，影响了业务的健康发展和生产规模的扩大。同时由于信息难以共享，使得企业间交易协作的效率较为低下，比如由于票据流通不便导致的合同结算慢的问题。

区块链可以实现核心企业的应收应付等信息在供应商、经销商、金融机构间流转，传递企业的信用，帮助参与企业解决融资方面的难题。区块链在实现这些企业之间共享交易数据的同时，可有效保护商业隐私和信息安全。此外，区块链的智能合约可实现企业债券流转、资金自动清算，加速企业的业务运作和资金流转的效率。

**场景四：电子政务**

传统的身份、所有权、社保记录等信息存在容易被篡改，信息更新不及时问题，容易被利用进行欺诈；归集部门承担保证数据有效性的职责，但缺乏必要的技术手段；数据归集部门与使用部门间存在信任断点问题，难以进行全面归集。归集部门承担最大的数据安全职责，如何避免批量导出的隐私泄露和解决数据利用过程的可管理可追溯问题，存在安全和效率的矛盾。

基于区块链智能合约定义联盟方共同协商一致达成的共识逻辑，只能有明确的政务事项权责的部门才能请求访问共享账本数据，并都

记录日志，以便能追溯。同时通过智能合约定义的加解密流程，确保数据防泄露；通过联盟方多方背书保证社会证照数据可信。

**场景五：互助保险**

这种模式是基于特定人群自发组织成一个保险风险池互帮互助，无中心信任背书。此时，如何提高保险资金的公平合理使用就显得尤为重要。

利用区块链技术公示所有保险资金的流向，增加了松散群体之间的信任，能帮助这种保险模式取得更好的发展前景。而且，能够提升互助计划的透明度，增加会员的信任度。同时，还可以赋能机构提升自律披露水平，更好地配合监管。

**场景七：物联网**

物联网设备将数据共享到业务网络专用区块链分类帐本。通过物联网区块链服务，您可以使用实时设备数据来实现供应链中交付的流程或产品的合规性，合同验证和协作等。优化了多方合作的信任关系，高效合作。

## 1.5 使用限制

- **Hyperledger Fabric 版本支持**

服务网络目前支持 Hyperledger Fabric 1.4.3 版本。关于 Fabric 版本信息请参考：

https://hyperledger-fabric.readthedocs.io/en/release-1.4/

- **节点资源访问限制**

服务网络城市节点是多应用共享使用的区块链资源。为保证其安全和稳定性,不支持用户直接访问城市节点底层资源,如记账节点、排序节点、CA 集群、应用链码容器、账本数据库和文件系统等。

- **应用通道管理功能**

出于安全性考虑，用户只能通过城市节点网关开放的 API 进行业务交易和应用通道块信息查询等业务类操作,未来将逐步开放应用通道相关的管理和配置功能。

- **与外部 Fabric 节点互联**

服务网络暂不支持用户外部 Fabric 节点与城市节点互联。本功能将在服务网络正式商用后逐步开放。

- **账本数据迁移**

目前不支持区块链账本数据迁移。用户可开发应用程序将账本数据从原有区块链写入到服务网络新账本中。

- **链码开发**

服务网络暂不支持应用链码在线开发和调试环境。未来服务网络将提供有限制的应用测试环境以便于应用在线调试和测试。

## 1.6 更新说明（bsnbase.com）

| | |
|---|---|
| 发布时间 | 2019/11/06 |
| 版本号 | 1.0.1 |
| 更新内容 | 1. 增加用户操作引导，用户首次登录时，系统提示用户查看在线帮助和开发实例；<br>2. 在我发布的服务页面新增【公开申请】功能，服务发布后可以申请公开服务，公开的服务将显示在应用商店中；<br>3. 在我发布的服务页面新增【参与应用】功能，服务发布成功以后，点击可以参与自己发布的服务；<br>4. 服务发布时增加了部分环节的邮件提醒功能；<br>5. 修复了部分已知的 BUG，增强了系统的稳定性。 |

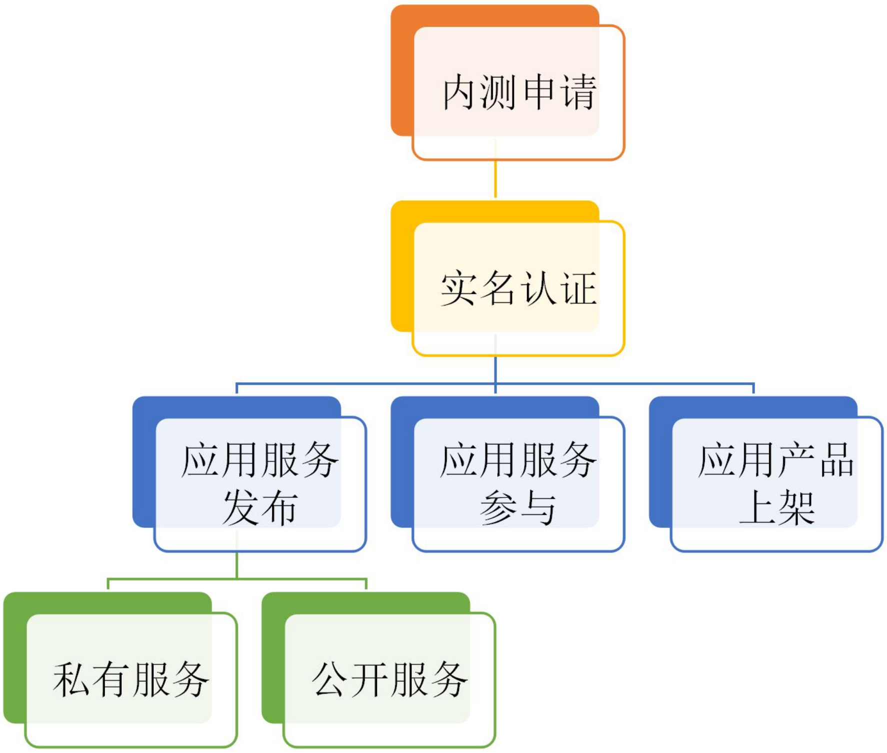
## 1.7 示例应用

参见：4.6 示例应用

## 1.8 开发资源

参见：5 开发资源

# 2 快速入门




- **内测申请**

登录服务网络门户网址：www.bsnbase.com，进入内测申请页面，按照页面提示输入相应的信息即可发起内测申请；内测申请须经后台用户的审核，审核通过后系统将向申请者发送激活邮件，申请者激活后即可登录门户系统。

- **实名认证**

用户登录门户系统后，默认为已实名认证。可以在服务网络中发布应用及参与其他用户发布的应用服务。

- **应用服务参与**

登录用户可以在应用商店中选择公开的服务并申请参与，申请后等待发布者审核通过即可正常使用。

- **应用产品上架**

应用产品是指在服务网络门户应用商店中上架的只能购买的应用；在用户购买之后，产品开发者为购买者单独发布。登录用户录入产品信息后即可申请上架，经运营系统审核通过之后显示在应用商店中。

- **应用服务发布**

应用服务是指在区块链服务网络中部署并运行的区块链应用。登录用户在页面中录入服务基本信息、链码信息、功能角色及发布的城市节点信息等即可申请发布服务，经运营系统审核、测试、复审、发布等流程后即可正式投入使用。

- **私有服务**

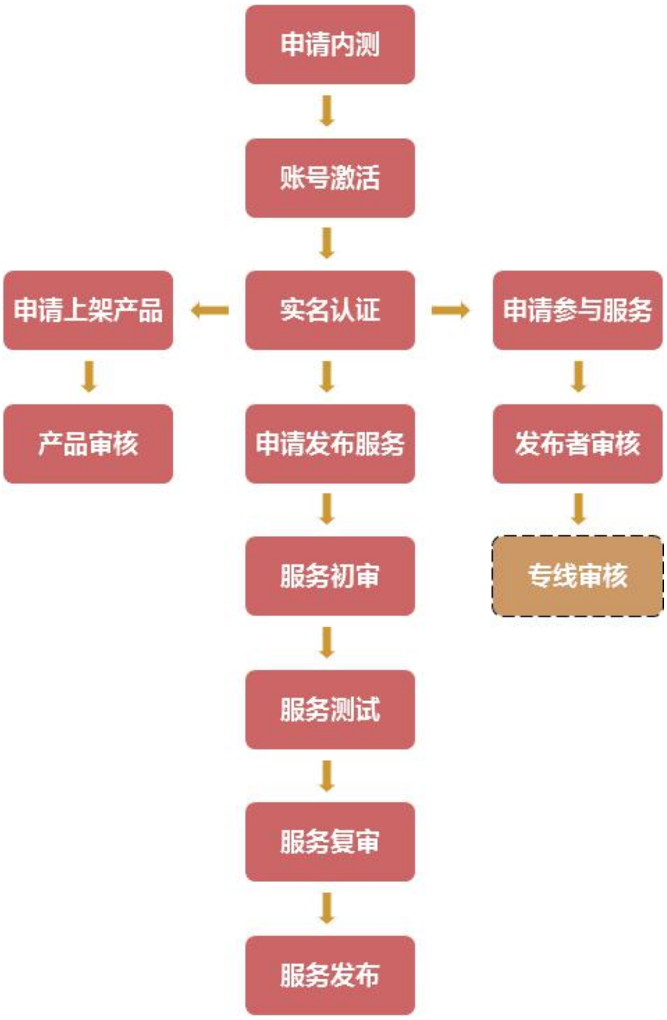
私有服务无法在服务网络门户中参与或搜索到，服务发布者可以通过邀请的方式来邀请服务网络的其他用户参与服务，或在其链下业务系统内向参与者用户提供应用服务。内测期间发布的服务默认是私有服务，如要公开，需另行申请。

- **公开服务**

公开服务可以在服务网络门户中搜索到，门户中其他用户可以自行申请参与。

# 3 用户指南

## 3.1 总体流程




用户使用区块链服务网络门户系统前，应先申请系统内测账户，申请成功之后由系统管理员进行核审；若系统管理员审核通过，则系统将向申请者发送账号激活邮件；用户需打开邮件中的激活链接完成账号激活。用户账号激活成功之后可登录系统。

内测期间登录的用户默认已完成了实名认证，可以执行系统的功能操作。

本系统的主要操作分为三个部分：产品上架、服务发布和服务参与。其中：

- **产品上架：**用户可在系统中根据要求录入产品信息并申请上架产品，申请成功之后需系统运营人员审核，审核通过之后即可上架到应用商店，详见3.4应用产品上架；
- **服务发布：**用户可在系统中根据要求录入服务信息并申请发布服务，申请成功之后需经系统运营人员初审、测试、复审、发布之后即可上架到应用商店，详见3.5应用服务发布；
- **服务参与：**用户可在应用商店中选择服务并申请参与；申请成功之后经服务发布者审核通过即参与成功。但若申请参与服务时，使用了专线功能则除了发布者审核外还需要运营人员审核才可参与服务。

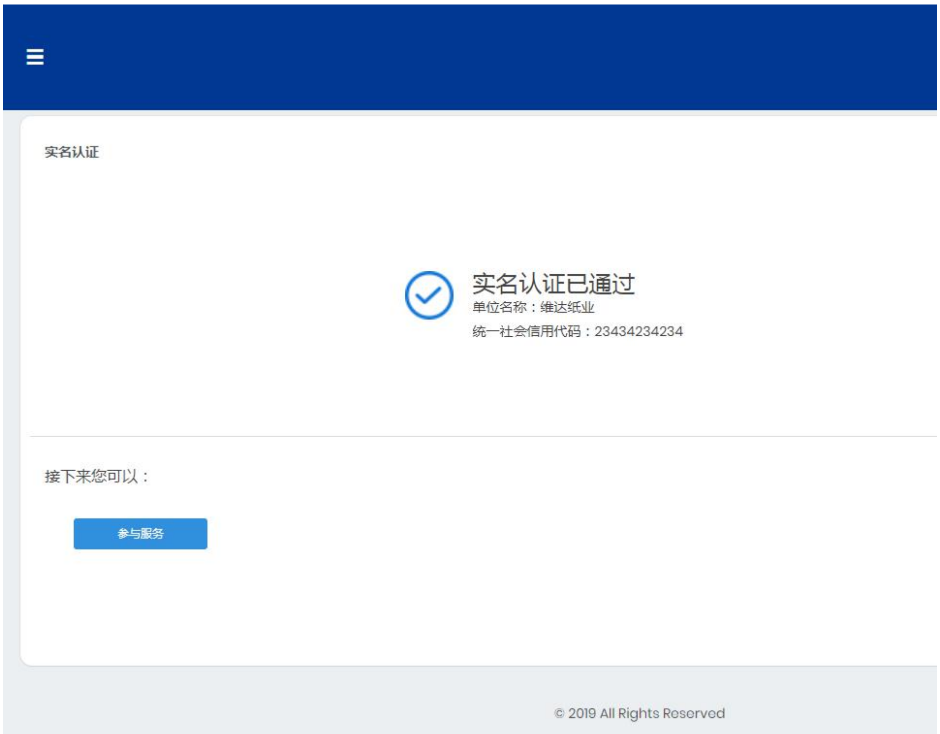
## 3.2 内测申请和激活

- 打开网络地址：www.bsnbase.com；
- 点击登录页面中的【内测申请】按钮进入内测申请页面；


- 根据页面中的提示填写内测申请信息并提交内测申请，内测申请用户分为【个人用户】和【企业用户】两种，申请人应根据自己的用户类型任选一种进行申请；
- 内测申请提交后需等待系统审核，待系统审核通过后将向申请者发送激活邮件，申请用户可根据【激活邮件】中的激活链接来激活账户；
- 帐户激活后即可登录系统使用。

## 3.3 实名认证

区块链应用的发布、上架和参与都需在实名认证通过之后才可以进行操作。内测申请审核通过的用户默认为【实名认证】已通过，用户登录【区块链服务网络门户】后，可在【用户中心】->【实名认证】页面查看实名认证信息。




## 3.4 应用产品上架

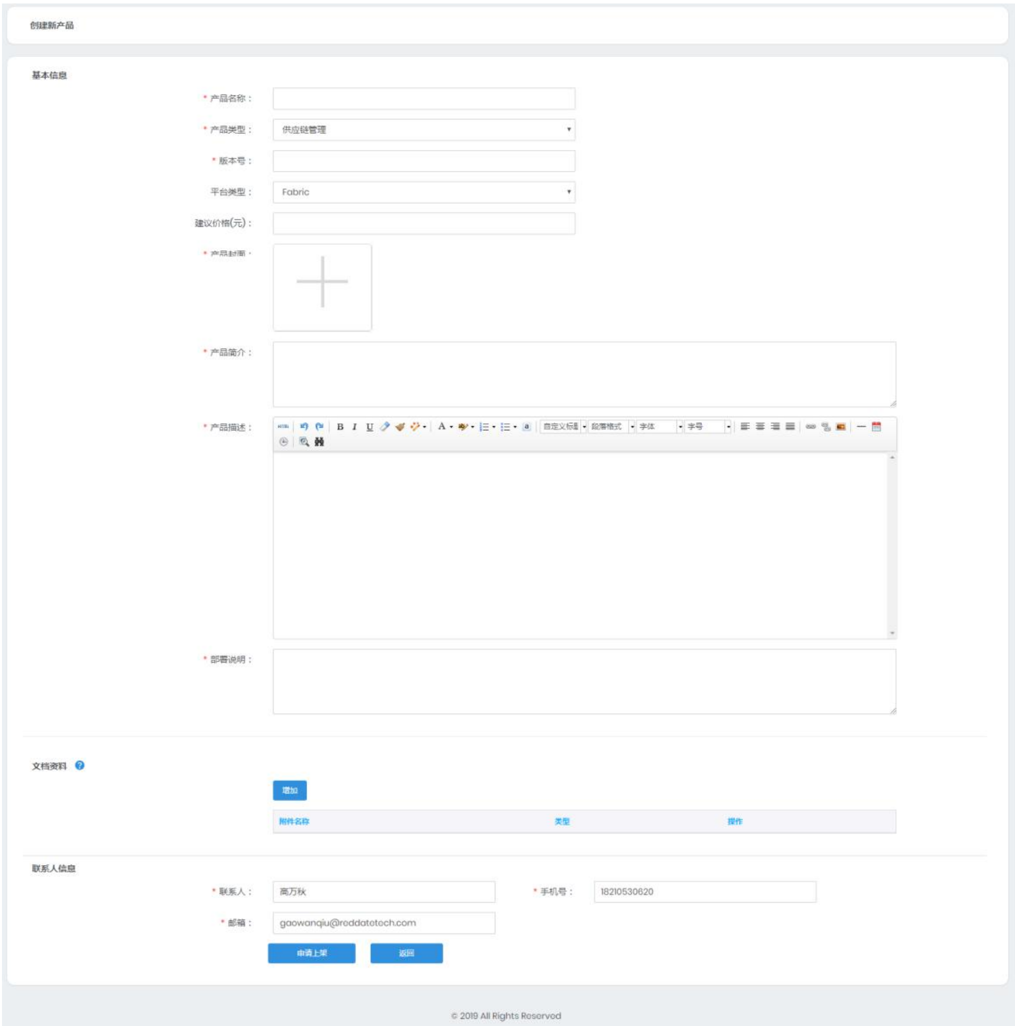
应用产品指没有在区块链服务网络中发布和运行，仅在服务网络门户的应用商店内上架的区块链应用。产品销售后，由购买方在服务网络上发布后方可以投入使用。产品发布完成之后即成为服务，此时购买者成为应用发布者，并可以邀请其他用户参与。产品的设置是让某些开发者仅以开发和销售产品为商业模式，而不需要投入成本去运营自己的产品(对比互联网，产品开发者类似提供网站搭建的开发者，而服务发布者类似经营网站的运营者)。

用户实名认证通过以后，可以申请上架产品。

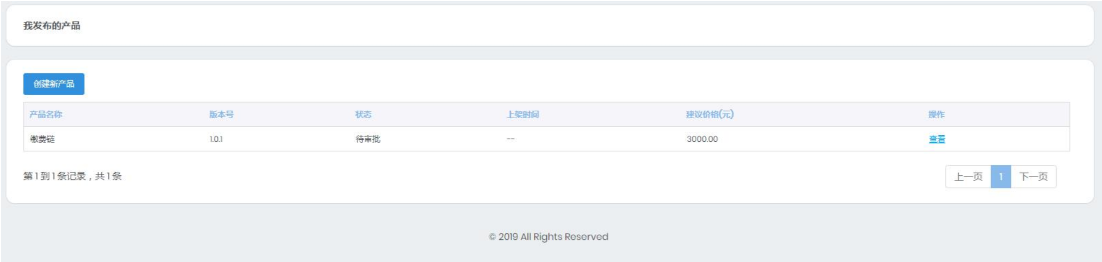
应用产品上架操作步骤如下：

- 在【我发布的应用】->【我发布的产品】页面进行产品的发布：


● 点击创建新产品，进入创建新产品页面，根据提示输入相应的信息：




● 点击【申请上架】，申请上架成功后，在【我发布的应用】->【我发布的产品】页面显示该条新创建的产品，新创建的产品状态为“待审批”；




- 待审批的产品经审核通过后上架到应用商店中。用户可在【应用商店】或【我发布的应用】->【我发布的产品】页面可以查看到新上架的产品。

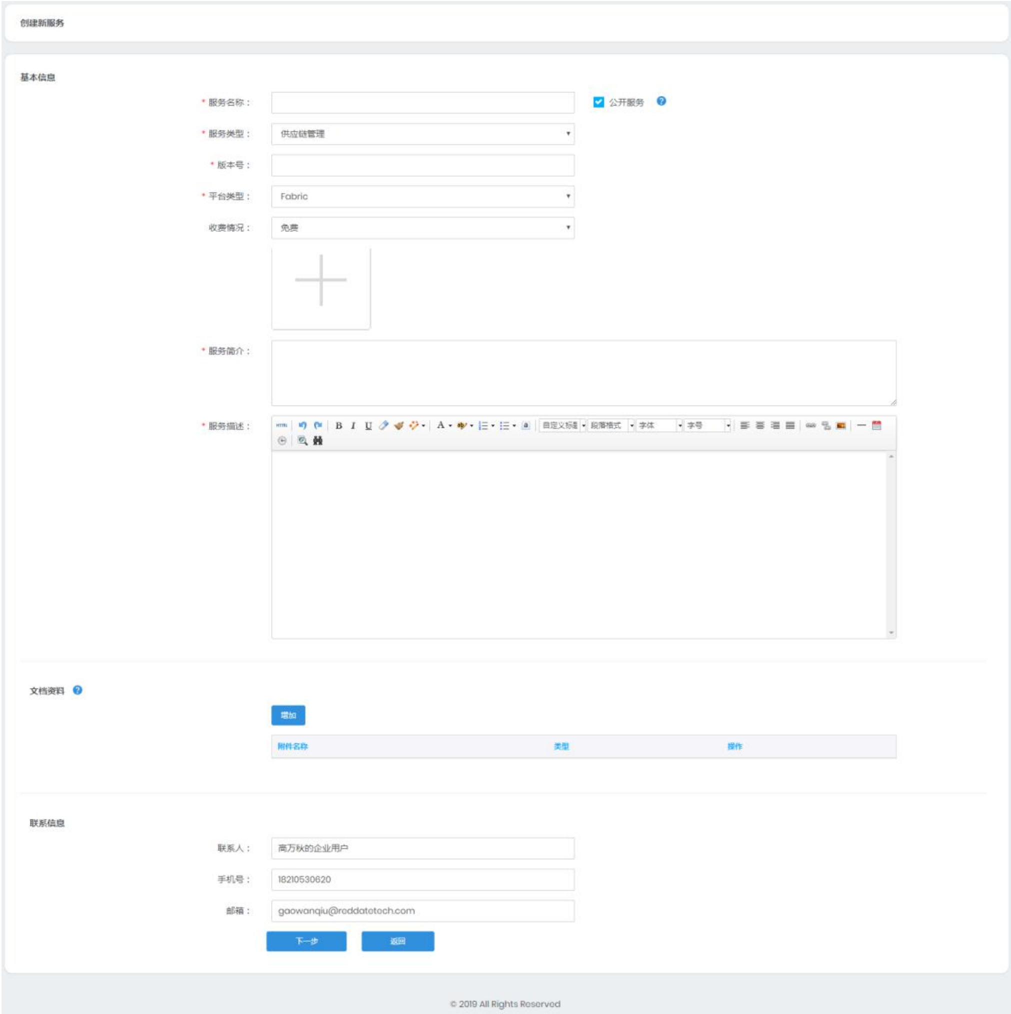
## 3.5 应用服务发布

应用服务是指在区块链服务网络中已经发布并运行的区块链应用，用户可以通过服务网络门户或发布方自己的业务系统直接申请参与使用。服务分为公开服务和非公开服务；用户在服务网络门户中无法搜索到非公开的服务。内测期间发布的服务默认为非公开服务。

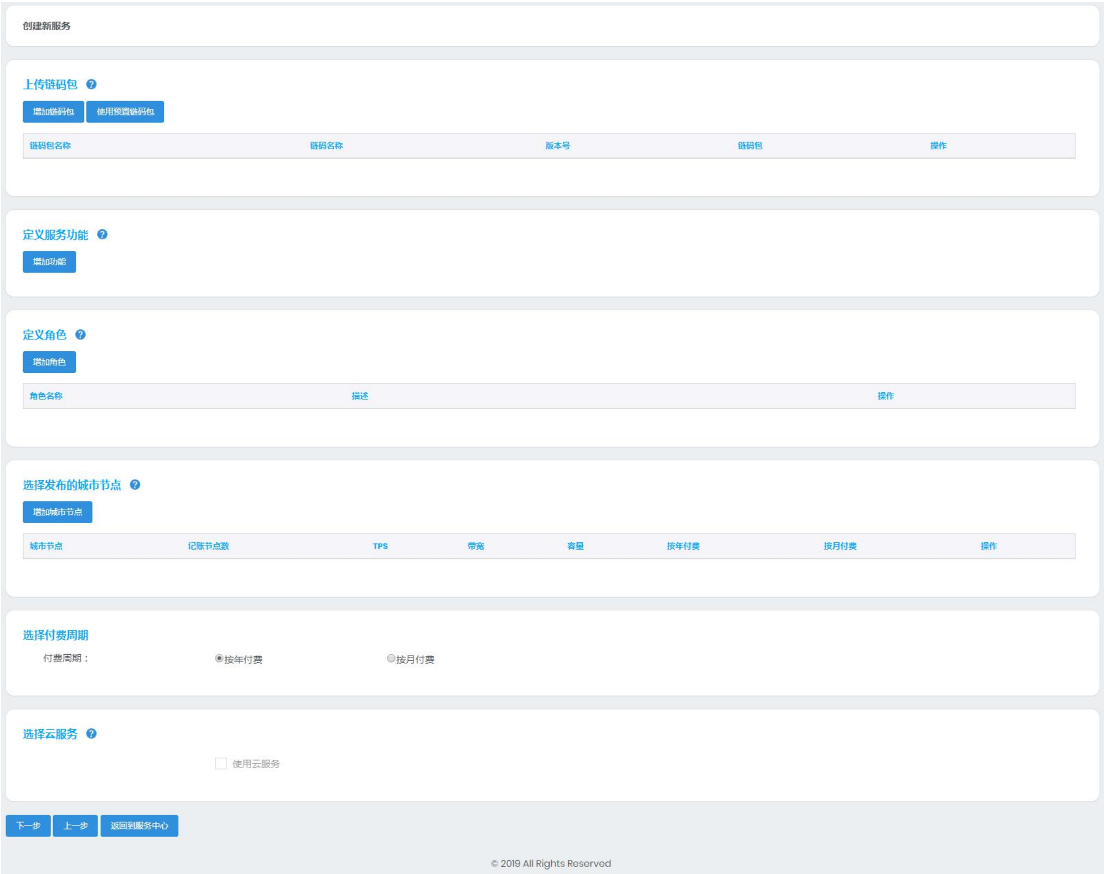
操作步骤如下：

- 点击【我发布的应用】->【我发布的服务】页面发布服务；


- 点击创建新服务，进入创建新服务页面，根据提示输入相应的信息：




- 点击【下一步】，开始上传链码包，定义服务功能及角色、选择发布的城市节点、付费周期以及是否使用云服务等信息：



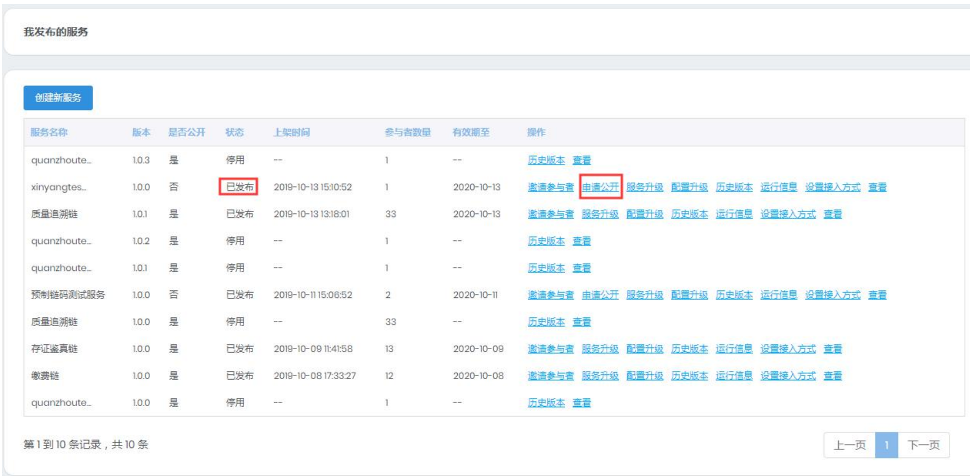

- 继续点击【下一步】，进入到【服务账单】页面，如下：


- 点击【确定】后，在【我发布的应用】->【我发布的服务】页面可以看到该条新创建的服务，且该服务的状态为“待初审”，发布者需等待后台进行审核和发布。服务发布后发布者默认不参与服务。

## 3.6 服务公开申请

内测期间发布的服务默认是非公开服务，用户若要公开服务，则需要另行申请。操作步骤如下：

- 在【首页】->【我发布的应用】->【我发布的服务】页面选择已发布且未公开的服务，点击【申请公开】，如下图：



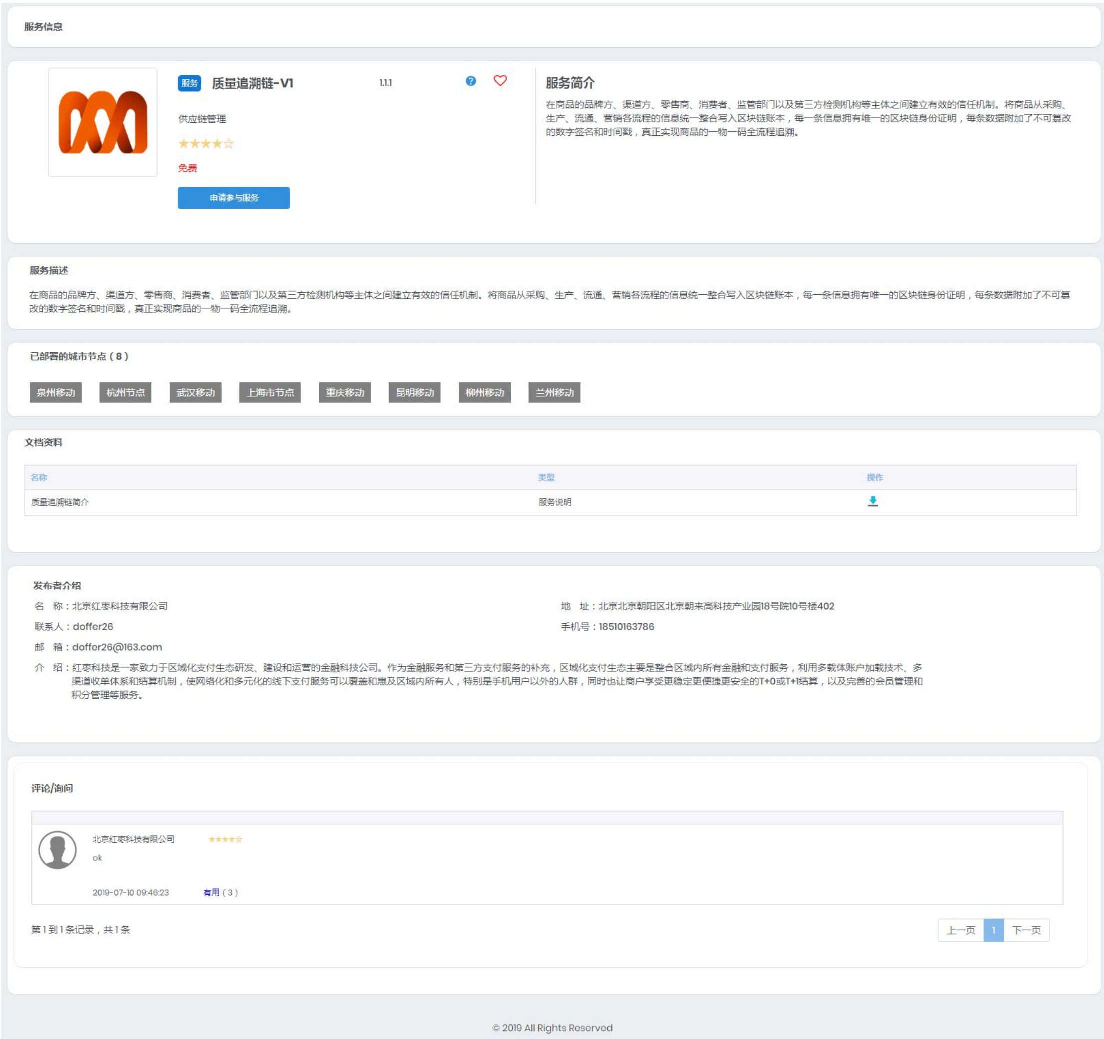

- 在系统弹出的提示中点击【确定】即可申请公开服务，如下图：


- 用户申请公开服务后，需经运营人员审核通过方可公开，公开的服务将显示在应用商店中。

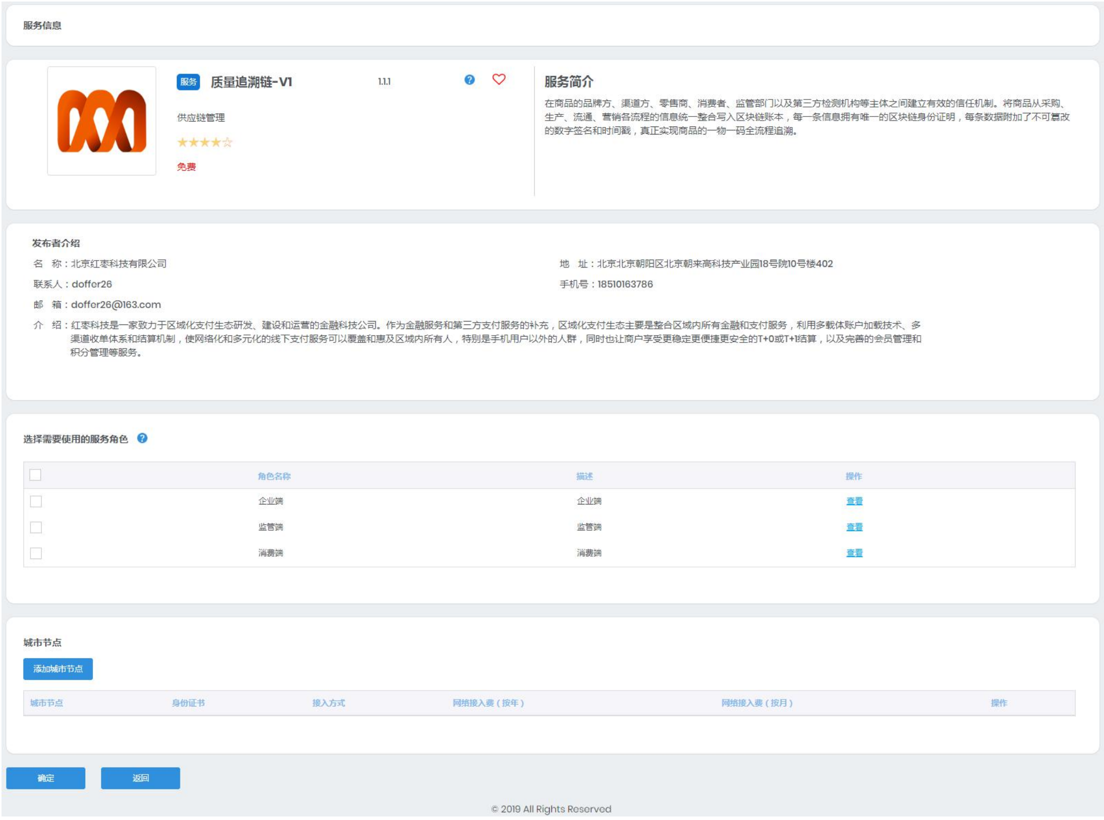
## 3.7 应用服务参与

用户实名认证通过后，即可参与区块链服务网络中所有已发布并公开的服务。操作步骤如下：

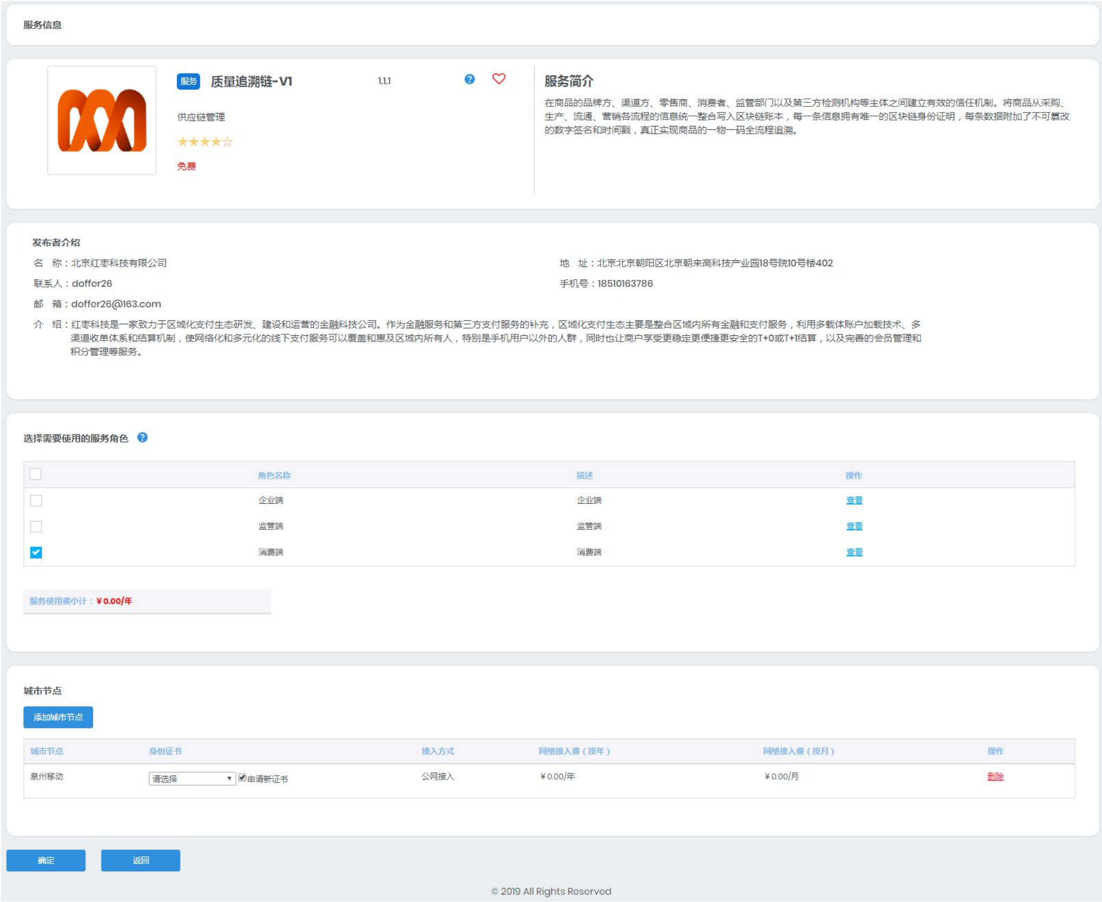
- 在【首页】->【我参与的应用】->【我参与的】页面或【应用商店】，选择一个未参与的应用去参与：




- 点击【申请参与服务】，进入选择应用角色及接入方式页面：




- 选择需要使用的应用角色和城市节点（角色是指用户参与服务所需使用的功能；城市节点是指用户从哪个城市来参与该服务）。用户选择角色后，系统会显示角色对应的功能使用费；



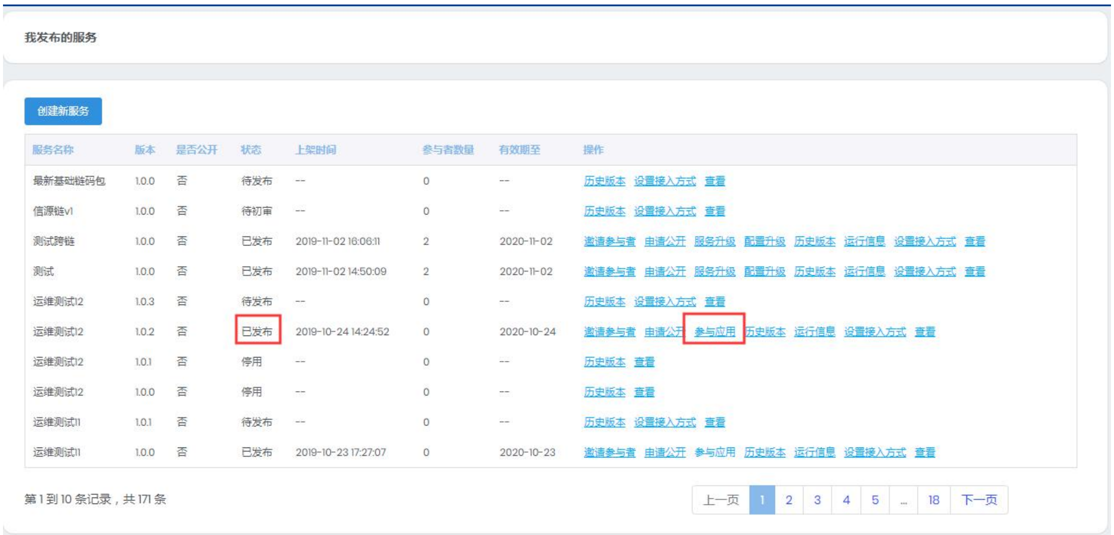

● 点击【确定】，提示接入城市节点、接入方式以及费用信息：


● 确认参与服务的接入城市节点、接入方式以及费用信息（如金额不为 0，还需进行支付）后，在【我参与的应用】->【我参与的】页面可以查看到所参与的应用服务，该应用服务的状态为“待发布者审核”，等待应用服务的发布者对参与者进行审核。


- 发布者和网络运营方对服务参与申请审核通过后，服务参与信息的状态更新为“已确认”。至此，参与应用服务成功。


注意：非公开的服务无法自由申请参与，只能通过发布者主动邀请其他用户参与。

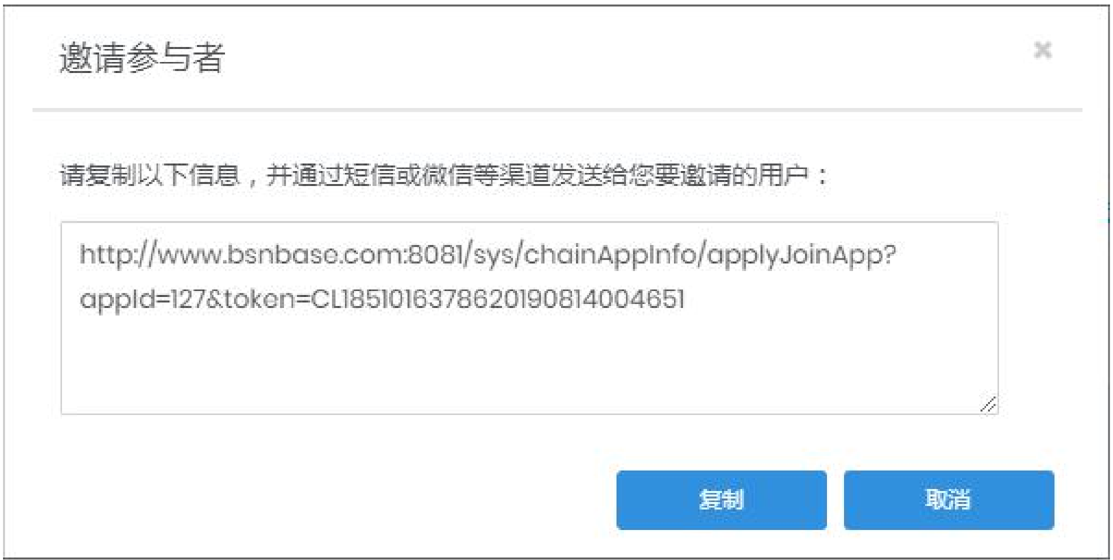
- 服务发布后，发布者默认不参与自己发布的服务，如需参与，则可以在【我发布的服务】列表中点击【参与应用】来参与自己发布的服务（具体步骤同上述应用服务参与步骤），如下图：




## 3.8 应用服务管理

服务发布成功之后，服务发布者还可以对服务的日常运维进行管理，包括邀请参与者、服务升级、配置升级、续费等。

- 邀请参与者：邀请区块链服务网络中其他用户参与服务；
  - 服务发布者登录之后进入【我发布的应用】->【我发布的服务】页面；
  - 选择状态为【已发布】的服务，点击【邀请参与者】，显示如下：




  - 点击【复制】即可复制邀请链接，并可以将邀请链接通过短信或微信等渠道发送给被邀请的用户；被邀请的用户点击链接并登录，可在区块链服务网络门户中参与此服务；
- 服务升级：开发者完成应用服务功能升级开发后，可在门户发起应用服务升级申请，申请需要经过后台的初审、测试、复审等流程，在服务升级发布完成之前原有版本继续正常运行，升级发布完成后自行切换并运行新版本；
  - 服务发布者登录之后进入【我发布的应用】->【我发布的服务】页面；
  - 选择状态为【已发布】的服务，点击【服务升级】，进入

服务升级页面：

- ■ 服务发布者通过修改应用的基本信息、文档资料、联系信息、链码信息、服务功能、服务角色等信息并支付相应的费用来完成应用升级；

## 3.9 应用服务计费

应用服务使用费包括应用资源使用费和应用功能使用费。功能使用费指服务参与者参与区块链服务时向服务发布者支付的功能使用费；应用资源使用费是指服务发布者发布区块链服务时向门户运营系统支付的TPS、带宽、容量等资源的使用费。

服务发布时，运营系统将服务需要的资源提交给计费系统，由计费系统计算和返回资源使用费。

服务参与时，运营系统将向计费系统提交应用功能、专线等信息，由计费系统计算和返回相应的费用。

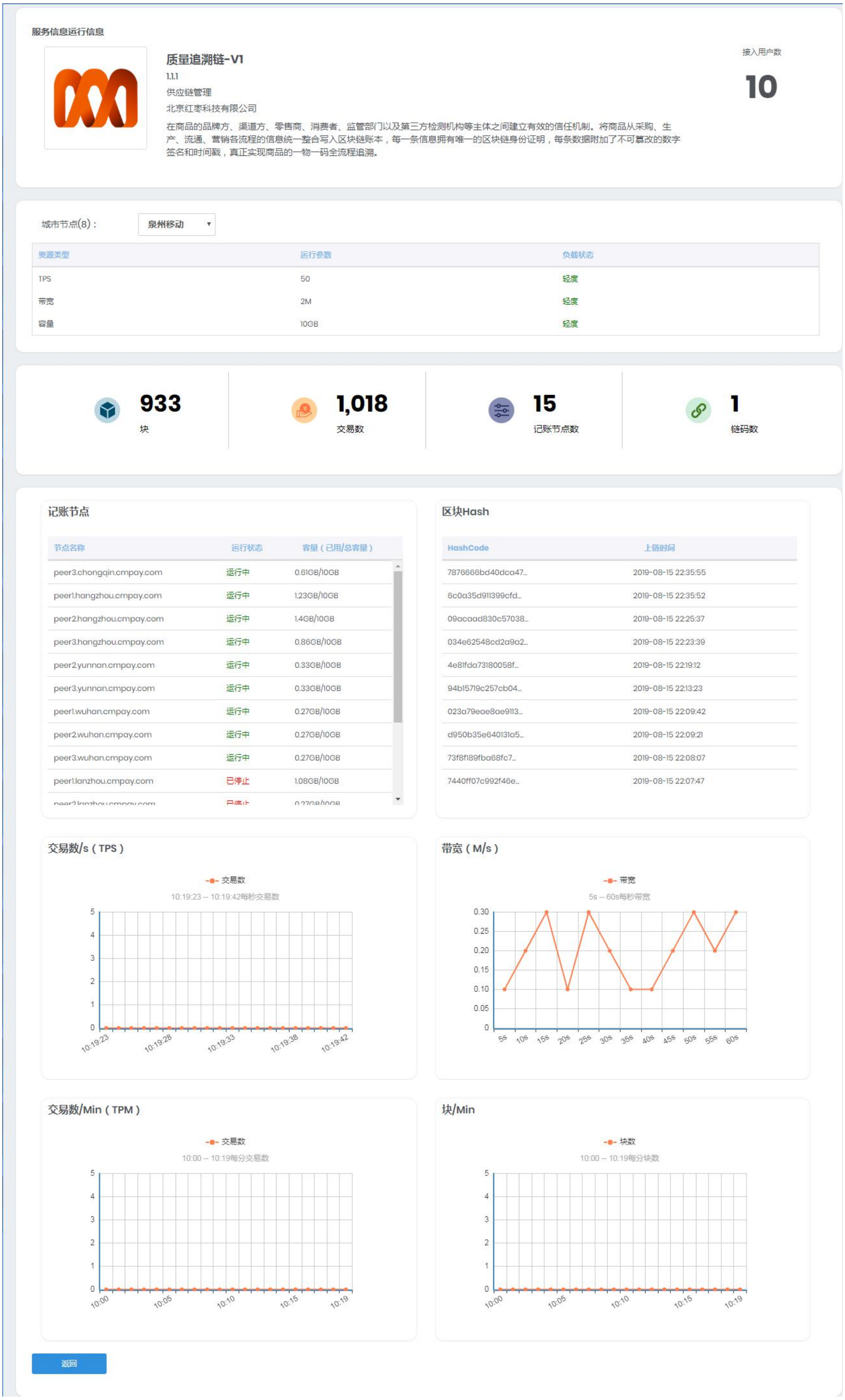
服务网络在内测期间，每位参与内测的开发者可以使用三个城市节点的资源免费发布一个应用服务。详细说明请参考服务网络官网的内测说明。

## 3.10 应用服务运行监控

应用服务运行监控主要负责收集、处理、汇总系统中的运行状态数据，并以可视化方式将系统的实时量化数据呈现出来。

- 在【我发布的服务】页面中，选择状态为【已发布】的服务，

点击【运行信息】可以查看当前服务的运行信息，页面如下：

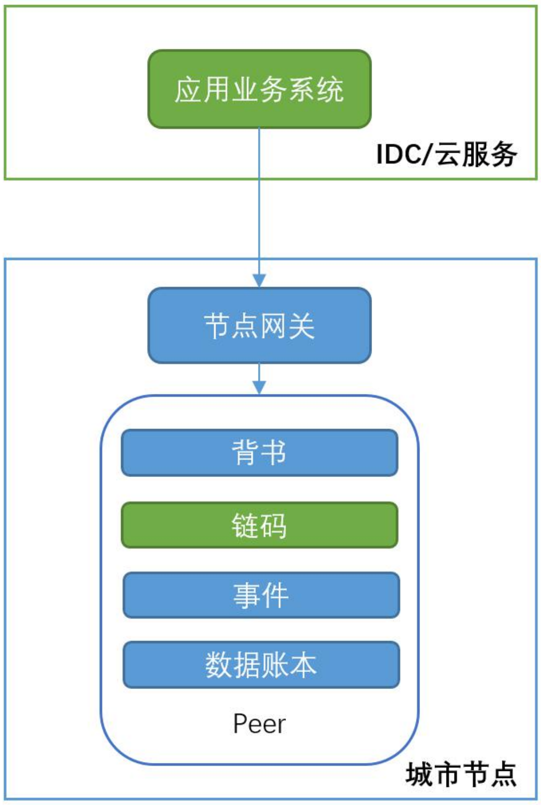

服务信息运行信息
质量追溯链-V1
1.1.1
供应链管理
北京红枣科技有限公司
在商品的品牌方、渠道方、零售商、消费者、监管部门以及第三方检测机构等主体之间建立有效的信任机制。将商品从采购、生产、流通、营销各流程的信息统一整合写入区块链账本，每一条信息拥有唯一的区块链身份证明，每条数据附加了不可篡改的数字签名和时间戳，真正实现商品的一物一码全流程追溯。
接入用户数
10
城市节点(8)：
泉州移动
资源类型 运行参数 负载状态
TPS 50 轻度
带宽 2M 轻度
容量 10GB 轻度
933
块
1,018
交易数
15
记账节点数
1
链码数
记账节点
节点名称 运行状态 容量（已用/总容量）
peer3.chongqin.cmpay.com 运行中 0.61GB/10GB
peer1.hangzhou.cmpay.com 运行中 1.23GB/10GB
peer2.hangzhou.cmpay.com 运行中 1.4GB/10GB
peer3.hangzhou.cmpay.com 运行中 0.86GB/10GB
peer2.yunnan.cmpay.com 运行中 0.33GB/10GB
peer3.yunnan.cmpay.com 运行中 0.33GB/10GB
peer1.wuhan.cmpay.com 运行中 0.27GB/10GB
peer2.wuhan.cmpay.com 运行中 0.27GB/10GB
peer3.wuhan.cmpay.com 运行中 0.27GB/10GB
peer1.lanzhou.cmpay.com 已停止 1.08GB/10GB
区块Hash
HashCode 上链时间
7876666bd40dca47… 2019-08-15 22:35:55
6c0a35d911399cfd… 2019-08-15 22:35:52
09acaad830c57038… 2019-08-15 22:25:37
034e62548cd2a9a2… 2019-08-15 22:23:39
4e81fda73180058f… 2019-08-15 22:19:12
94b15719c257cb04… 2019-08-15 22:13:23
023a79eae8ae9113… 2019-08-15 22:09:42
d950b35e640131a5… 2019-08-15 22:09:21
73f8f189fba68fc7… 2019-08-15 22:08:07
7440ff07c992f46e… 2019-08-15 22:07:47
交易数/s（TPS）
交易数
10:19:23 – 10:19:42每秒交易数
带宽（M/s）
带宽
5s – 60s每秒带宽
交易数/Min（TPM）
交易数
10:00 – 10:19每分交易数
块/Min
块数
10:00 – 10:19每分块数
返回

# 4 开发指南

## 4.1 概述

区块链服务网络（BSN）面向政企、事业单位以及开发者提供基于联盟链技术和共识信任机制的区块链技术基础设施服务，以互联网理念为客户提供公共区块链资源环境，帮助客户快速、低成本的开发、部署、运营和维护区块链应用服务。

区块链应用开发一般包含应用智能合约（应用服务链码）和应用系统（即链下的业务系统）两部分。链下业务系统通过服务网络城市节点网关调用在节点上部署的应用服务链码进行链上交易处理、数据写链和数据查询等操作。服务发布者和参与者可使用 IDC 资源或云服务平台部署自己的链下业务系统，并通过互联网或专线连接到城市节点网关访问应用服务链码。




### 4.1.1应用产品和服务

在区块链服务网络 bsnbase.com 中，应用包括应用服务和应用产品两类，其中：

- **应用服务**是指在区块链服务网络中部署并运行的区块链应用。应用发布者可以在服务网络门户中发布自己的区块链应用服务，并根据应用的运营模式将服务设置为公开服务或非公开服务。服务网络的用户可以在应用商店查看并申请参与公开服务；非公开服务不会出现在应用商店中，只能通过发布者邀请的方式参与。应用发布者负责应用服务参与申请的审批并管理参与者的应用功能权限。
- **应用产品**是指仅在服务网络门户的应用商店内上架而没有在

服务网络中发布和运行的区块链应用。服务网络的用户可以在应用商店查看应用产品详细信息，与产品开发者达成购买协议后，由购买方在服务网络上发布该产品的应用服务。产品发布完成之后即成为服务，购买者成为应用服务发布者，并可以邀请其他用户参与。产品的设置是让某些开发者仅以开发和销售产品为商业模式，而不需要投入成本去运营自己的产品。

### 4.1.2身份和权限验证

在服务网络中，每个应用参与者在业务接入的城市节点都拥有独立的身份证书。证书由区块链服务网络托管，用户可以在服务门户下载自己的证书。用户可以使用在同一城市节点的已有证书参与该城市节点的多个应用，也可以为不同应用申请新的证书。

链下业务系统访问链上的数据时，应当使用自己的用户证书或者用户 CODE 通过节点网关进行链上交易处理。服务发布者可以在服务网络门户为应用参与者设置应用的链码权限。该权限控制着参与者可以访问应用的哪些链码或者哪些链码的方法，如果参与者没有所要访问的链码或者链码方法的权限，访问将被拒绝。

开发应用的时候应当注意，在设置应用内部权限的时候应该根据应用的链码权限设置，否则可能出现参与者访问时拥有应用功能权限而没有该功能所对应的链码的权限的错误。

### 4.1.3数据安全和数据隐私

服务网络的区块链应用服务运行在各自相互隔离的交易通道上，其数据也存储在各自的数据库中，服务网络节点间采用 TLS 数据通讯保证数据传输安全。应用可以在链下业务系统数据上链时进行数据加密或进行 hash 处理，以保证链上交易处理和数据存储的数据安全。如果采用 hash 数据，服务发布者需要自行提供业务合作方间共享源数据的方法；如果采用加密数据，则需要在应用业务合作方间共享解密密钥的方法。应用也可以选择在应用链码中构建访问控制和加密处理，以控制应用参与者的数据访问权限和保证数据账本存储安全。

### 4.1.4应用交易处理性能

服务网络的目标是面向广泛的区块链行业应用交易处理提供基础设施服务，性能是实现这一目标的重要组成部分。应用性能主要由以下三个部分组成：

- 吞吐量：区块链每秒可处理的事务数；
- 延迟：客户端应用程序向服务提交交易事务与数据写入账本之间的时间（数据写链）；
- 容量：区块链存储交易数据的能力；

服务网络可同时支持每个应用每秒 1500 笔交易事务，事务提交到数据写链的延迟时间根据参与应用的城市节点和记账节点数量不同一般为 1~7 秒。未来我们将对服务网络的共识算法、交易背书、交易广播、数据账本更新、计算资源分配、网络传输等方面进行持续优

化增加服务网络的吞吐量。

从服务网络合作的各大云服务商（如：中国移动、中国电信、AWS等）那里我们可以持续不断地引入高性能、大存储容量的服务器、网络资源和存储资源，这些数量庞大的高质量计算资源、网络资源和存储资源将大幅度的提升服务网络的应用负载，以支撑越来越多的区块链应用运行。

### 4.1.5应用写链数据要求

过大的数据文件如图片或电子文档通过链上传输将会降低应用交易TPS，增大交易事务处理延时，建议单个交易事务的数据最大不超过4M。应用可以在链下共享业务组织间较大的源数据，在链上存储和共享源数据hash，链下业务系统使用链上源数据hash验证源数据的真实性和有效性。

## 4.2 应用链码开发

链码（ChainCode）又称为智能合约，是用计算机语言描述合约条款、交易的条件、交易的业务逻辑等，通过调用智能合约实现交易的自动执行和对账本数据的操作。一个区块链应用可以部署多个链码，每个链码包含多个方法。

链码支持多种语言编写，包括golang、java、node.js。每个链码程序都必须实现Chaincode接口，链码包含：Init ,Invoke,Query三个基本操作：

- **Init** ：链码初始化的方法，在链码实例化或者升级的时候调用一次，以便链码可以执行任何必要的初始化，包括应用程序状态的初始化。
- **Invoke** :接收和处理链下业务系统调用事务处理提案，其参数包含调用的链码程序中函数的名称和具体业务处理数据参数。即在 Invoke 中根据不同的方法参数调用其他分支处理响应的业务。Invoke 可以简单的理解为链码方法的入口。
- **Query**:提供查询链码数据的方法，该方法只作为查询使用，不提供操作链上数据的操作。可在 Query 操作时调用，亦可在 Invoke 方法中作为某些方法的分支被调用。该方法可以不实现。

### 4.2.1开发规范

为实现自动化部署，提升部署效率，针对不同语言的链码开发做出以下规范。

注:发布服务时,链码包打包时进入项目根目录进行打包,格式为.zip。

1、 GO

main 函数，必须在项目中所有链码的上级或同级

例: BsnBaseChainCode 链码包

```
BsnBaseChainCode
├──main.go
├──ChainCode/
```

```
├──models/
└──utils/
```

说明：main.go 入口，ChainCode：链码；models：实体；utils：工具类

2、 Java

gradle 或 maven 构建的项目

例：BsnBaseChainCode 链码包

```
BsnBaseChainCode
 │  build.gradle
 └──src
     └──main
         └──java
             └──com.example.javacc
                 └──javacc.java
```

说明：src/main/java：项目目录，com.example.javacc：包名，javacc.java：链码信息

3、 NodeJs

必须在项目根目录下创建 package.json

例：BsnBaseChainCode 链码包

```
BsnBaseChainCode
    │  marbles_chaincode.js
    └──package.json
```

说明：marbles_chaincode.js：链码

注:发布服务时,链码包打包时进入项目根目录进行打包,格式为.zip。

### 4.2.2预置链码包

预置链码包是我们为应用开发者提供对业务数据进行增删改查基本操作的链码（Golang 语言编写）。应用开发者可以在此链码包的基础上根据应用业务需求进一步拓展链码功能。此链码支持存储的数据类型有字符串、整型、浮点型、集合（map、list）等等。

链码包下载地址：[www.bsnbase.com/static/base/BaseChainCode.zip](www.bsnbase.com/static/base/BaseChainCode.zip)

链码包功能如下：

1. 增加数据（set）

输入参数说明

baseKey：需要保存的唯一的主键标识

baseValue：保存的数据信息

例：{"baseKey":"str","baseValue":"this is string"}

其中 baseKey 是不能为空的字符串，baseValue 可以是任意类型的数据。如果 baseKey 已经存在，则直接返回已经存在，不能添加；如果不存在，则添加数据。

2. 修改数据（update）

输入参数说明

baseKey：需要修改的唯一的主键标识

baseValue：保存的数据信息

例：{"baseKey":"str","baseValue":"this is string"}

其中 baseKey 是不能为空的字符串，baseValue 可以是任意类型的数据。如果 baseKey 不存在，则无法更新，如果已经存在，则修改数据。

3. 删除数据（delete）

输入参数说明

baseKey：需要删除的唯一的主键标识的值

例："str"

其中 baseKey 的值不能为空，且必须存在，否则将无法删除。

4. 获取数据（get）

输入参数说明

baseKey：需要获取的唯一的主键标识的值

例："str"

其中 baseKey 的值不能为空，且必须存在，否则将无法获取到相应的信息。

5. 获取历史记录数据（getHistory）

输入参数说明

baseKey：需要获取的唯一的主键标识的值

例："str"

其中 baseKey 的值不能为空。响应结果：交易 Id（txId）、交易时间（txTime）、是否删除（isDelete）、交易信息（dataInfo）。

欢迎开发者们共享自己的通用链码作为服务网络预置链码包，与

我们一起拓展服务网络的区块链应用支撑能力。

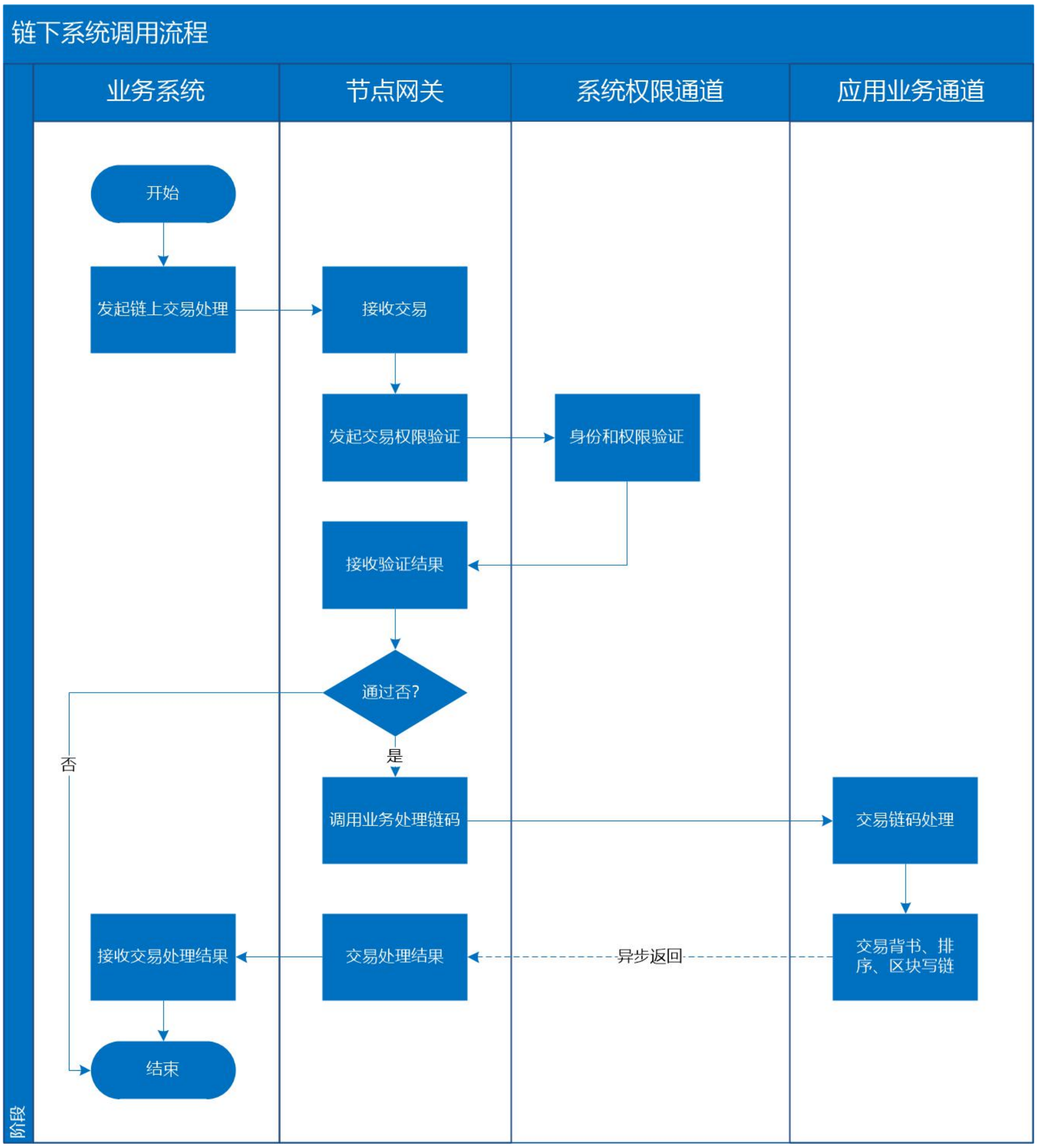
## 4.3 应用服务发布

应用链码开发完成后，开发者可在在服务网络门户发布和部署应用服务，并参与应用服务获取应用链下业务系统开发所需的节点网关访问地址、应用服务参数、身份证书等必需的应用服务配置信息。请参考《区块链服务网络帮助手册》的应用服务发布和应用服务参与章节的说明，完成应用服务发布和参与。

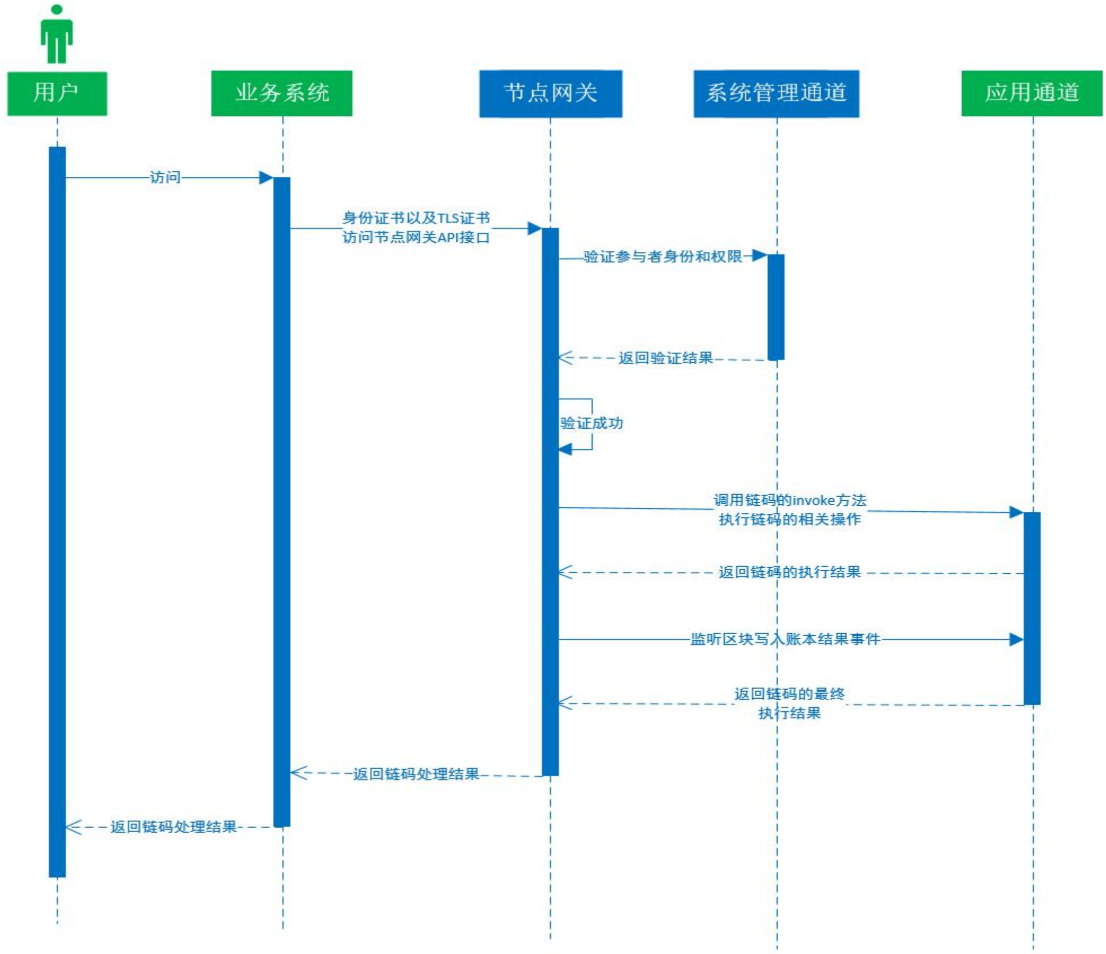
## 4.4 应用系统开发

开发者需要自行开发应用的链下业务系统的业务逻辑，开发过程中可通过节点网关 API 来调用应用链码执行交易事务处理、完成数据写链和数据查询等操作。

# 4.4.1应用交易处理流程



### 4.4.2应用交易处理时序




## 4.5 节点网关 API

节点网关是部署在各个城市节点，接收应用系统的请求，使用托管的用户身份信息，向相应的应用链码发起访问并返回链码的执行结果。节点网关的调用是通过向区块链服务的各个城市节点的网关服务发送 HTTP 请求来实现。节点网关负责验证用户身份信息、应用信息，通过用户身份信息和应用信息以及需要访问的链码、链码方法来传递链码参数、获取链码执行结果的服务通道。

### 4.5.1调用应用链码

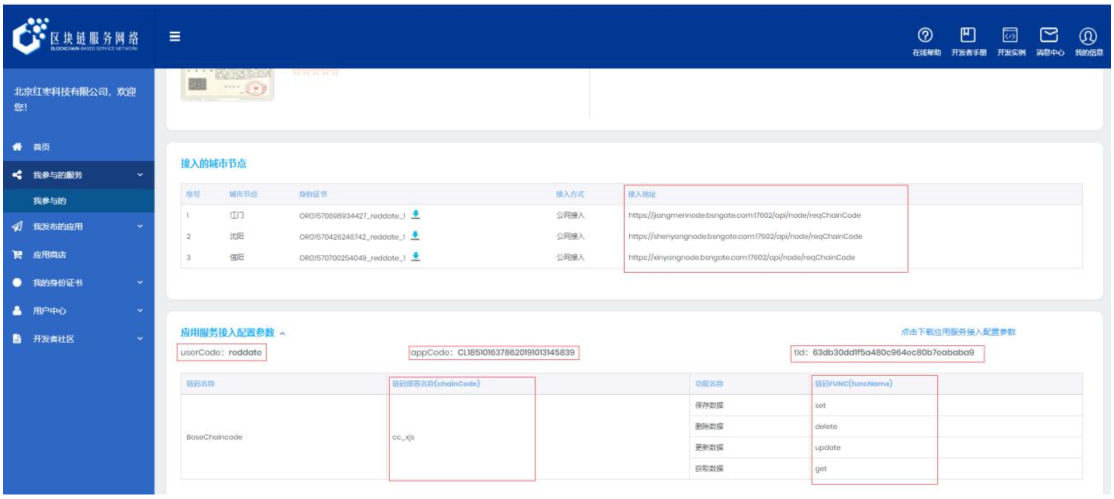
业务系统需要按照接口说明在请求中加入相应的请求参数，调用节点网关以后，节点网关会返回链码的执行结果。

1. 接口地址：https://节点网关地址/api/node/reqChainCode

   或者 https://节点网关地址/api/node/syncChainCode

   前者不等交易落块结果直接响应结果，可以调用根据交易 ID 查询落块结果；后者等待交易落块后响应结果（响应相对前者会慢），可根据自身业务需求自行选择。

注：用户参与服务成功后可以在服务详情页面查看并下载应用链下业务系统开发所需要的应用服务配置参数、节点网关地址和应用身份证书，如下图：



2. 通讯方式： POST
3. 签名算法：
   - 将 userCode+ appCode+ chainCode+ funcName 的值以及 args 中每一项数据拼接成字符串 A；
   - 对字符串 A 使用用户证书的私钥进行 SHA256WITHECDSA 签

名。

## 4. 请求参数

| 序号 | 字段名 | 字段 | 类型 | 必填 | 备注 |
|---|---|---|---|---|---|
| 1 | 信息头 | header | Map | 是 | |
| 2 | 信息体 | body | Map | 是 | |
| 3 | 签名值 | mac | String | 是 | |
| **header** | | | | | |
| 1 | 用户唯一标示 | userCode | String | 是 | |
| 2 | 应用唯一标识 | appCode | String | 是 | |
| 3 | 用户与应用 ID | tId | String | 否 | |
| **body** | | | | | |
| 1 | 链码 Code | chainCode | String | 是 | |
| 2 | 方法名称 | funcName | String | 是 | |
| 3 | 请求参数 | args | String[] | 否 | |
| 4 | 临时数据 | transientData | Map<string,string> | 否 | |

注：签名值：加密对象：userCode+ appCode+ chainCode+ funcName+args[0]+args[1]+…+args[args.length-1]
私钥：通过证书下载获取的私钥
用户与应用 ID：用户参与应用后的关联 ID
请求参数：是否为空由具体的业务场景决定，可以为空

示例：


```
{
    "header":{
        "appCode":"CL1881038873220190902114314",
        "userCode":"newuser",
        "tId":""
    },
    "body":{
        "args":[
            "16399b5085ee5d3981f5076c33c5a0a66d7f2f3545b4d88501116a8bd53d13a5",
            "{"fileName":"test.jpg","fileHash":"6cfacf57e5b27f71a47a812938021784"}"
        ],
        "funcName":"set",
        "chainCode":"cc_bcj"
    },

"mac":"MEUCIQDTFe2Gerdf7YJrG1a1Yt99M0ZQ3T1lGpsXdNmFV7WuTgIgSkZ19abUhAJbMrJMBoD8N7f26xhpQRuR4vNAfY7EEbs="
}
```


签名值：

newuserCL1881038873220190902114314cc_bcjset16399b5085ee5d3981f5076c33c5a0a66d7f2f3545b4d88501116a8bd53d13a5{"fileName":"test.jpg","fileHash":"6cfacf57e5b27f71a47a812938

021784"}

## 5. 响应参数

| 序号 | 字段名 | 字段 | 类型 | 必填 | 备注 |
|---|---|---|---|---|---|
| 1 | 信息头 | header | Map | 是 | |
| 2 | 信息体 | body | Map | 是 | |
| 3 | 签名值 | mac | String | 是 | |
| **header** | | | | | |
| 1 | 响应标识 | code | int | 是 | 0：校验成功<br>-1：校验失败 |
| 2 | 响应信息 | msg | String | 否 | code=0 是可为 null |
| **body** | | | | | |
| 1 | 块信息 | blockInfo | Map | 否 | code 不为 0 时为空 |
| 2 | 链码响应结果 | ccRes | Map | 否 | code 不为 0 时为空 |
| **blockInfo** | | | | | |
| 1 | 交易 Id | txId | String | 是 | |
| 2 | 状态值 | status | Int | 是 | 详见下表 6、响应状态描述 |
| **ccRes** | | | | | |
| 1 | 链码响应状态 | ccCode | Int | 是 | 200：成功<br>500：失败 |
| 2 | 链码响应结果 | ccData | Str | 否 | 具体链码响应的结果 |

示例


```
{
    "header":{
        "code":0,
        "msg":"处理成功"
    },
    "body":{
        "blockInfo":{
          "txId":"62ef371fe90cda1586c6757b924aaa48cbd9d2ec057325ebf30df052e8fa6134",
            "status":0
        },
        "ccRes":{
            "ccCode":200,
            "ccData":"保存版权信息成功"
        }
    },
"mac":"MEQCID27XE05k2gN71s2R94CfWZ79L6BG7daN3uDNvzBnk4IAiACR/05PKJMDA
```

| |
|---|
| HpOurMTjC1KGiHeeZKhmdARU47CIUelg=="<br>} |
| Mac 验签<br>在门户中下载证书时，会下载相应网关的公钥证书，需要使用该证书验签<br>签名值：<br>newuserCL18810388732201909021143l4cc_bcjset16399b5085ee5d3981f5076c33c5a0a66d7f2f3545b4d88501116a8bd53d13a5{"fileName":"test.jpg","fileHash":"6cfacf57e5b27f71a47a812938021784"} |

6. 响应状态描述

| 序号 | 状态码 | 备注 |
|---|---|---|
| 1. | 0 | 成功<br>等待响应结果时，落块的最终状态<br>不等待响应结果：链码的处理状态 |
| 2. | -1 | 等待落块结果超时 |
| 3. | 1 | 空的提交数据 |
| 4. | 2 | 异常的响应 |
| 5. | 3 | 错误的提交信息 |
| 6. | 4 | 错误的创建者签名 |
| 7. | 5 | 无效的“背书人”交易 |
| 8. | 6 | 无效的事务配置 |
| 9. | 7 | 不支持的交易响应 |
| 10. | 8 | 错误的交易 ID |
| 11. | 9 | 重复的交易 ID |
| 12. | 10 | 背书失败 |
| 13. | 13 | 未知的交易类型 |
| 14. | 14 | 找不到目标链码 |
| 15. | 17 | 链码过期 |
| 16. | 18 | 链码版本冲突 |
| 17. | 254 | 无效的交易 |
| 18. | 255 | 其他原因的无效交易 |

### 4.5.2获取块信息

数据上链后，节点网关会响应当前交易信息（body.blockInfo），状态值(body.blockInfo.status)和交易 Id(body.blockInfo.txId)，如果状态值为 0 时，表示交易提交成功并且落块，可根据交易 Id 查询块信息。

1. 接口地址：https://节点网关地址/api/node/getBlockInfo
2. 通讯方式： POST
3. 签名算法：
   - 将 userCode+ appCode+ txId 的值拼接成字符串 A；
   - 对字符串 A 使用用户证书的私钥进行 SHA256WITHECDSA 签名。
4. 请求参数

| 序号 | 字段名 | 字段 | 类型 | 必填 | 备注 |
|---|---|---|---|---|---|
| 1 | 信息头 | header | Map | 是 | |
| 2 | 信息体 | body | Map | 是 | |
| 3 | 签名值 | mac | String | 是 | |
| **header** | | | | | |
| 1 | 用户唯一标示 | userCode | String | 是 | |
| 2 | 应用唯一标识 | appCode | String | 是 | |
| **body** | | | | | |
| 1 | 交易 Id | txId | String | 是 | |
| 注：签名值：加密对象：userCode+ appCode+ txId<br>私钥：通过证书下载获取的私钥<br>用户与应用 ID：用户参与应用后的关联 ID | | | | | |
| 示例： | | | | | |
| {"header":{"appCode":"CL1881038873220190902114314","userCode":"newuser"},"body":{"txId":"d0024e8421bece167f2531009760d9aa81dcae26e45daa0c7daf1fb3c5f4e3e4"},"mac":"MEQCIBRhaM2szckWl9N9qcqnaYXOXGQw7SfII9DlRvxcI3YVAiBt4XeNs+EUjhBNSr3IjLRPZucsuGHxfjt9RiaNIQS8cA=="} | | | | | |
| 签名值：<br>newuserCL1881038873220190902114314 | | | | | |

5. 响应参数

| 序号 | 字段名 | 字段 | 类型 | 必填 | 备注 |
|---|---|---|---|---|---|
| 1 | 信息头 | header | Map | 是 | |
| 2 | 信息体 | body | Map | 是 | |
| 3 | 签名值 | mac | String | 是 | |
| **header** | | | | | |
| 1 | 响应标识 | code | int | 是 | 0：校验成功<br>-1：校验失败 |

| | | | | | |
|---|---|---|---|---|---|
| 2 | 响应信息 | msg | String | 否 | code=0 时可为 null |
| **body** | | | | | |
| 1 | 块信息 | blockInfo | Map | 否 | code 不为 0 时为空 |
| 2 | 链码响应结果 | ccRes | Map | 否 | 为空 |
| **blockInfo** | | | | | |
| 1 | 块 Hash | blockHash | String | 是 | |
| 2 | 块号 | blockNumber | Long | 是 | |
| 3 | 上一个块 Hash | preBlockHash | String | 是 | |
| 4 | 块大小 | blockSize | Long | 是 | 字节 |
| 5 | 当前块的交易数量 | blockTxCount | Int | 是 | |
| 示例 | | | | | |


```
{
    "header": {
        "code": 0,
        "msg": "处理成功"
    },
    "body": {
        "blockInfo": {
            "blockHash":
"92c5fe7aac0c5a8c04b7a8671c2af0a9c7cf8875aa4ece5a09eee7e074be67ea",
            "blockNumber": 11,
            "preBlockHash":
"a5e65c2c317dd721cb36be37e7b35c77ae25e56c93cdec3e46c95e530caaab68",
            "blockSize": 7006,
            "blockTxCount": 1
        },
        "ccRes": null
    },
    "mac":
"MEQCIHpN5OAKscZxavtFIt7wKcFEZTZxgWFpXA4PK0vlAl8+AiAHM96NVUxOFqnlFmr
LlY3q+3pNgRDZw/WlI+OQjOkzoQ=="
}
```


Mac 验签

在门户中下载证书时，会下载相应网关的公钥证书，需要使用该证书验签

签名值：与请求时签名值相同

newuserCL1881038873220190902114314

### 4.5.3获取最新账本信息

获取当前应用最新的账本信息，包含块 Hash，上一个块 Hash，

当前块的快高等信息。

1. 接口地址：https://节点网关地址/api/node/getLedgerInfo
2. 通讯方式：POST
3. 签名算法：
   - 将 userCode+ appCode 的值拼接成字符串 A；
   - 对字符串 A 使用用户证书的私钥进行 SHA256WITHECDSA 签名。
4. 请求参数

| 序号 | 字段名 | 字段 | 类型 | 必填 | 备注 |
|---|---|---|---|---|---|
| 1 | 信息头 | header | Map | 是 | |
| 2 | 信息体 | body | Map | 是 | |
| 3 | 签名值 | mac | String | 是 | |
| **header** | | | | | |
| 1 | 用户唯一标示 | userCode | String | 是 | |
| 2 | 应用唯一标识 | appCode | String | 是 | |
| 注：签名值：加密对象：userCode+ appCode<br>私钥：通过证书下载获取的私钥 | | | | | |
| 示例： | | | | | |
| {"header":{"appCode":"CL1881038873220190902114314","userCode":"newuser"},"body":{},"mac":"MEQCIBRhaM2szckWl9N9qcqnaYXOXGQw7SfII9DlRvxcI3YVAiBt4XeNs+EUjhBNSr3IjLRPZucsuGHxfjt9RiaNIQS8cA=="} | | | | | |
| 签名值：<br>newuserCL1881038873220190902114314 | | | | | |

5. 响应参数

| 序号 | 字段名 | 字段 | 类型 | 必填 | 备注 |
|---|---|---|---|---|---|
| 1 | 信息头 | header | Map | 是 | |
| 2 | 信息体 | body | Map | 是 | |
| 3 | 签名值 | mac | String | 是 | |
| **header** | | | | | |
| 1 | 响应标识 | code | int | 是 | 0：校验成功<br>-1：校验失败 |

| | | | | | |
|---|---|---|---|---|---|
| 2 | 响应信息 | msg | String | 否 | code=0 时可为 null |
| **body** | | | | | |
| 1 | 块信息 | blockInfo | Map | 否 | code 不为 0 时为空 |
| **blockInfo** | | | | | |
| 1 | 块 Hash | blockHash | String | 是 | |
| 2 | 块高 | height | Long | 是 | |
| 3 | 上一个块 Hash | preBlockHash | String | 是 | |
| 示例 | | | | | |


```
{
    "header": {
        "code": 0,
        "msg": "处理成功"
    },
    "body": {
        "blockInfo": {
            "blockHash":
"92c5fe7aac0c5a8c04b7a8671c2af0a9c7cf8875aa4ece5a09eee7e074be67ea",
            " height ": 11,
            "preBlockHash":
"a5e65c2c317dd721cb36be37e7b35c77ae25e56c93cdec3e46c95e530caaab68"
        },
        "ccRes": null
    },
    "mac":
"MEQCIHpN5OAKscZxavtFIt7wKcFEZTZxgWFpXA4PK0vlAl8+AiAHM96NVUxOFqnlFmr
LlY3q+3pNgRDZw/WlI+OQjOkzoQ=="
}
```


Mac 验签
在门户中下载证书时，会下载相应网关的公钥证书，需要使用该证书验签
签名值：与请求时签名值相同
newuserCL1881038873220190902114314

## 4.5.4链码事件注册

参与的服务中，如果有链码事件，需要提醒业务系统进行下一步处理时，可调用该接口注册监听的链码事件。

1. 接口地址： https:// 节点网关地址

/api/chainCode/event/register

2. 通讯方式： POST

3. 签名算法：

■ 将 userCode+ appCode+ chainCode+ eventKey+ notifyUrl+attachArgs 的值拼接成字符串 A;

■ 对字符串 A 使用用户证书的私钥进行 SHA256WITHECDSA 签名。

4. 请求参数

| 序号 | 字段名 | 字段 | 类型 | 必填 | 备注 |
|---|---|---|---|---|---|
| 1 | 信息头 | header | Map | 是 | |
| 2 | 信息体 | body | Map | 是 | |
| 3 | 签名值 | mac | String | 是 | |
| **header** | | | | | |
| 1 | 用户唯一标示 | userCode | String | 是 | |
| 2 | 应用唯一标识 | appCode | String | 是 | |
| **body** | | | | | |
| 1 | 链码 Code | chainCode | String | 是 | |
| 2 | 链码事件 key | eventKey | String | 是 | |
| 3 | 链码事件通知地址 | notifyUrl | String | 是 | 接收监听到的链码事件的地址 |
| 4 | 附加参数 | attachArgs | String | 否 | |
| 示例： | | | | | |
| {"header":{"appCode":"CL20191107112252","userCode":"lessing"},"body":{"attachArgs":"name=张三 &age=20","chainCode":"cc_bsn_test_00","eventKey":"test01","notifyUrl":"http://192.168.6.128:8080/api/event/notifyUrl"},"mac":"MEUCIQCjzPr4KZVild2Vm5YgcunOXTh9mQK2QfWcRnYCk+jOzgIgDW6oHca7/249M43p2ElwiMNbuejdwAnyW5OwiMqiWCQ="} | | | | | |

5. 响应参数

| 序号 | 字段名 | 字段 | 类型 | 必填 | 备注 |
|---|---|---|---|---|---|
| 1 | 信息头 | header | Map | 是 | |
| 2 | 信息体 | body | Map | 是 | |

| | | | | | |
|---|---|---|---|---|---|
| 3 | 签名值 | mac | String | 是 | |
| **header** | | | | | |
| 1 | 响应标识 | code | int | 是 | 0：注册成功<br>-1：注册失败 |
| 2 | 响应信息 | msg | String | 是 | |
| **body** | | | | | |
| 1 | 事件编码 | eventId | String | 是 | |
| 示例 | | | | | |
| { "header": { "code": 0, "msg": "注册链码事件成功" }, "body": { "eventId": "bd3391deedbe44a7ad5b7f80ce59abfa" }, "mac": "MEQCIENLpj2R9mRL100vcMXs0X5rwfSjB/U7kMg+76GjEPNJAiBlUo/Eyj49uXTPrzRW0m4rJ0NQIkZnDMPbyalxojXwrA=="} | | | | | |

## 4.5.5链码事件查询

用于查询自己注册的链码事件列表

1. 接口地址： https://节点网关地址/api/chainCode/event/query
2. 通讯方式： POST
3. 签名算法：
   - 将 userCode+ appCode 的值拼接成字符串 A;
   - 对字符串 A 使用用户证书的私钥进行 SHA256WITHECDSA 签名。
4. 请求参数

| 序号 | 字段名 | 字段 | 类型 | 必填 | 备注 |
|---|---|---|---|---|---|
| 1 | 信息头 | header | Map | 是 | |
| 2 | 信息体 | body | Map | 否 | |
| 3 | 签名值 | mac | String | 是 | |
| **header** | | | | | |
| 1 | 用户唯一标示 | userCode | String | 是 | |
| 2 | 应用唯一标识 | appCode | String | 是 | |
| 示例： | | | | | |
| {"header":{"appCode":"CL20191107112252","userCode":"lessing"},"body":{},"mac":"MEQCIAnJxvuKVe0u/bG0VYCjM3g3ctxTYIWkejYp462okNlcAiBcOTGvAkF7xErL2w1PiwgfFjIu3Ssz | | | | | |

gyfzym/pEwRGxA=="}

## 5. 响应参数

| 序号 | 字段名 | 字段 | 类型 | 必填 | 备注 |
|---|---|---|---|---|---|
| 1 | 信息头 | header | Map | 是 | |
| 2 | 信息体 | body | Map | 是 | |
| 3 | 签名值 | mac | String | 是 | |
| **header** | | | | | |
| 1 | 响应标识 | code | int | 是 | 0：查询成功<br>-1：查询失败 |
| 2 | 响应信息 | msg | String | 是 | |
| **body** | | | | | |
| 1 | 事件编码 | eventId | String | 是 | |
| 2 | 链码事件 key | eventKey | String | 是 | |
| 3 | 链码事件通知地址 | notifyUrl | String | 是 | |
| 4 | 附加参数 | attachArgs | String | 否 | |
| 5 | 创建时间 | createTime | String | 是 | |
| 6 | 用户唯一标示 | userCode | String | 是 | |
| 7 | 应用唯一标识 | appCode | String | 是 | |
| 8 | 链码 Code | chainCode | String | 是 | |
| 9 | 城市编码 | orgCode | String | 是 | |

示例

{ "header": { "code": 0, "msg": "查询链码事件成功" }, "body": [ { "eventKey": "test001", "notifyUrl": "http://192.168.6.128:8080/api/event/notifyUrl", "attachArgs": "a=123\u0026b=456", "eventId": "945ee631d26140118963ad3104c81713", "createTime": "2019-11-18 14:22:59", "orgCode": "ORG1571365934172", "userCode": "lessing", "appCode": "CL20191107112252", "chainCode": "cc_bsn_test_00" }, { "eventKey": "test002", "notifyUrl": "http://192.168.6.128:8080/api/event/notifyUrl", "attachArgs": "hahahhahhahahahah", "eventId": "346617a493d84c6d8512b8dddad87811", "createTime": "2019-11-18 14:29:28", "orgCode": "ORG1571365934172", "userCode": "lessing", "appCode": "CL20191107112252", "chainCode": "cc_bsn_test_00" }, { "eventKey": "test01", "notifyUrl": "http://192.168.6.128:8080/api/event/notifyUrl", "attachArgs": "name=张三\u0026age=20", "eventId": "bd3391deedbe44a7ad5b7f80ce59abfa", "createTime": "2019-11-19 10:52:15", "orgCode": "ORG1571365934172", "userCode": "lessing", "appCode": "CL20191107112252", "chainCode": "cc_bsn_test_00" } ], "mac": "MEQCIEYXFMa8dfBrjy/s9H5JAoFIrjROJBiw+7/daELUbF5eAiA7a6HvqqbOpv6vlkunHGxCB1o5DoeuJFD0FM6kLoU34Q=="}

### 4.5.6链码事件注销

注销注册的链码事件监听

1. 接口地址： https://节点网关地址/api/chainCode/event/remove
2. 通讯方式： POST
3. 签名算法：
   - 将 userCode+ appCode+ chainCode+ eventId 的值拼接成字符串 A;
   - 对字符串 A 使用用户证书的私钥进行 SHA256WITHECDSA 签名。
4. 请求参数

| 序号 | 字段名 | 字段 | 类型 | 必填 | 备注 |
|---|---|---|---|---|---|
| 1 | 信息头 | header | Map | 是 | |
| 2 | 信息体 | body | Map | 否 | |
| 3 | 签名值 | mac | String | 是 | |
| **header** | | | | | |
| 1 | 用户唯一标示 | userCode | String | 是 | |
| 2 | 应用唯一标识 | appCode | String | 是 | |
| **body** | | | | | |
| 1 | 链码 Code | chainCode | String | 是 | |
| 2 | 事件编码 | eventId | String | 是 | |
| 示例： | | | | | |
| {"header":{"appCode":"CL20191107112252","userCode":"lessing"},"body":{"eventId":"bd3391deedbe44a7ad5b7f80ce59abfa","chainCode":"cc_bsn_test_00"},"mac":"MEQCIE3/CLG5LxZZN7En7LZvzthajwxHzpvDduXSsw4Tb1JFAiAXGJ4WVtyCKbtCasQGofCkge8NOgZDNPgJIdTCtCi2SQ=="} | | | | | |

5. 响应参数

| 序号 | 字段名 | 字段 | 类型 | 必填 | 备注 |
|---|---|---|---|---|---|
| 1 | 信息头 | header | Map | 是 | |
| 2 | 信息体 | body | Map | 是 | |

| | | | | | |
|---|---|---|---|---|---|
| 3 | 签名值 | mac | String | 是 | |
| **header** | | | | | |
| 1 | 响应标识 | code | int | 是 | 0：注销成功<br>-1：注销失败 |
| 2 | 响应信息 | msg | String | 是 | |
| 示例 | | | | | |
| { "header": { "code": 0, "msg": "注销链码事件成功" }, "body": null, "mac": "MEUCIQCaTFLliY7pPjkwcmSsLXOth7k9bQj9Sblq+1nMVjkFAAIgUsizFO+f1+dxU3/hPxjf/+na4qG6aQFftJIWGtMhlVI="} | | | | | |

## 4.5.7链码事件通知报文信息

注册的链码事件监听到相应的链码事件后的通知报文信息。

服务器在收到请求后响应一个包含 success 的字符串，否则系统将分别在第 3、12、27、48 秒后再次发起通知，共计 5 次。

1. 通讯方式： POST
2. 签名算法：
   - 将 userCode+ appCode+ chainCode+ orgCode+ eventKey+ eventId+ eventName+ txId+ nonceStr 的值拼接成字符串 A；
   - 对字符串 A 使用用户证书的私钥进行 SHA256WITHECDSA 签名。
3. 参数

| 序号 | 字段名 | 字段 | 类型 | 必填 | 备注 |
|---|---|---|---|---|---|
| 1 | 信息头 | header | Map | 是 | |
| 2 | 信息体 | body | Map | 否 | |
| 3 | 签名值 | mac | String | 是 | |
| **header** | | | | | |
| 1 | 用户唯一标示 | userCode | String | 是 | |
| 2 | 应用唯一标识 | appCode | String | 是 | |
| **body** | | | | | |
| 1 | 链码 Code | chainCode | String | 是 | |

| | | | | | |
|---|---|---|---|---|---|
| 2 | 城市编码 | orgCode | String | 是 | |
| 3 | 注册的链码事件 key | eventKey | String | 是 | |
| 4 | 注册的链码事件 ID | eventId | String | 是 | |
| 5 | 注册的链码事件参数 | attachArgs | String | 否 | 注册时填写的附加参数 |
| 6 | 监听到的链码事件 key | eventName | String | 是 | 链码中设置的事件名称 |
| 7 | 监听到的链码事件值 | payload | String | 否 | |
| 8 | 当前链码的交易 Id | txId | String | 是 | |
| 9 | 当前交易块高 | blockNumber | Long | 是 | |
| 10 | 响应随机码 | nonceStr | String | 是 | 业务平台可根据该值判断是否接收过通知，同一个业务的多次通知中，该字符串不变 |

示例：

{"header":{"userCode":"lessing","appCode":"CL20191107112252"},"body":{"chainCode":"cc_bsn_test_00","orgCode":"ORG1571365934172","eventKey":"test:\\S{32}","eventId":"2964a0f60b3e460f834618b3664af2da","attachArgs":"abc=123211","eventName":"test:12345678123456781234567812345678","txId":"32fc105681820fa556b8a460efc1e43a47daa864b959ea1753abb4640f2dce49","payload":"","blockNumber":74,"nonceStr":"522c8061b5e84837bad72ca08c6a353f"},"mac":"MEQCIDU4tROyjLtvD1b8TTbWWAlCPuUbmdPAEUXwRRgVn7kIAiA58je5u/7xDuRPcgeUWL3nB9mouUGQ6dGKJMmD7Jm08g=="}

### 4.5.8加密算法

签名规则

- SHA256 哈希值计算：将需要签名的字符串做 Hash256 计算；
- 对哈希值获取签名：将 SHA 值与私钥进行椭圆曲线加密签名计算；

■ 将签名结果做 Base64 计算。

您可以在服务网络门户-文档资料中下载节点网关通讯报文数据签名与验签说明。

下载地址：www.bsnbase.com/sys/document/detail?docId=1

## 4.6 示例应用

如下是我们基于预制链码包开发的通用数据应用链下业务系统通过网关 API 调用链码的示例源代码，包括：golang、java、C#、python 语言的实例，供您下载参考。

下载地址(golang)：

www.bsnbase.com/static/base/BCCService-golang.zip

下载地址（java)：

www.bsnbase.com/static/base/BCCService-java.zip

下载地址(C#)：

www.bsnbase.com/static/base/BCCService-csharp.zip

下载地址(python)：

www.bsnbase.com/static/base/BCCService-py.zip

# 5 开发资源

➢ **新手篇：**

- 区块链是什么？分布式账本技术入门指南：

https://developer.ibm.com/cn/blog/2017/what-is-blockchain-a-primer-on-distributed-ledger-technology/?re=ibm_blockchain-cn-31

- 比特币与企业级区块链的区别：

https://developer.ibm.com/cn/tv/2017/the-difference-between-bitcoin-and-blockchain-for-business/?re=ibm_blockchain-cn-31

- 区块链技术基础：术语和用例：

https://www.ibm.com/developerworks/cn/cloud/library/cl-blockchain-basics-glossary-bluemix-trs/index.html?re=ibm_blockchain-cn-31

- 区块链技术基础：分布式账本简介：

https://www.ibm.com/developerworks/cn/cloud/library/cl-blockchain-basics-intro-bluemix-trs/?re=ibm_blockchain-cn-31

## ➢ 进阶篇

- 使用 Go 编写智能合约：

  https://www.ibm.com/developerworks/cn/cloud/library/cl-ibm-blockchain-chaincode-development-using-golang/index.html?re=ibm_blockchain-cn-31

- 面向 Java 开发人员的区块链链代码：

  https://www.ibm.com/developerworks/cn/java/j-chaincode-for-java-developers/index.html?re=ibm_blockchain-cn-31

- 使用 BPM 和区块链实现数字流程自动化：

  https://www.ibm.com/developerworks/cn/middleware/library/mw-1705-auberger-bluemix/1705-auberger.html?re=ibm_blockchain-cn-31

- 使用区块链和 IoT 执行灾难管理：

  https://developer.ibm.com/cn/blog/2018/disaster-management-using-blockchain-iot/?re=ibm_blockchain-cn-31

- 区块链技术指南：

  https://yeasy.gitbooks.io/blockchain_guide/

## ➢ Hyperledger Fabric 相关资料

- Hyperledger 文档（中文）：
  https://hyperledger-fabric-cn.readthedocs.io/zh/latest/
- Hyperledger Fabric 文档（英文）：
  https://hyperledger-fabric.readthedocs.io/en/release-1.4/

- Hyperledger 项目概览 社区介绍

  课程回放：

  https://mediacenter.ibm.com/media/1_kmn9xv78

  讲义下载：

  https://github.com/dWChina/ibm-opentech-ma/blob/master/fabric1.4LTS/02_Hyperledger_Overview.pdf

- Fabric 1.4 TLS 概述

  讲义下载：

  https://github.com/dWChina/ibm-opentech-ma/blob/master/fabric1.4LTS/03_Fabric_Architecture_Overview.pdf

  课程回放：

  https://mediacenter.ibm.com/media/0_yt6akmy3

- Fabric Peer 解析

  讲义下载：

  https://github.com/dWChina/ibm-opentech-ma/blob/master/fabric1.4LTS/04_HL_Fabric_Peer.pdf

  课程回放：

  https://mediacenter.ibm.com/media/0_bqrb4sha

- Fabric Order 解析

  讲义下载：

  https://github.com/dWChina/ibm-opentech-ma/blob/master/fabric1.4LTS/05_HL_Fabr ic_Orderer.pdf

课程回放：

https://mediacenter.ibm.com/media/0_we4e7el3

- Fabric MSP 与 CA

讲义下载：

https://github.com/dWChina/ibm-opentech-ma/blob/master/fabric1.4LTS/06_HL_Fabric_MSP_CA.pdf

课程回放：

https://mediacenter.ibm.com/media/0_38fba6u1

- Fabric 应用开发指南

讲义下载：

https://github.com/dWChina/ibm-opentech-ma/blob/master/fabric1.4LTS/07_HL_Fabric_App_Dev.pdf

课程回放：

https://mediacenter.ibm.com/media/0_jx5rgyd2

- Fabric 部署实践

讲义下载：

https://github.com/dWChina/ibm-opentech-ma/blob/master/fabric1.4LTS/08_HL_Fabric_Deployment.pdf

课程回放：

https://mediacenter.ibm.com/media/1_zoun8xuf

# 6 技术支持

- 邮箱：support@bsnbase.com

- 电话：400-071-8215（工作日8:00-17:30）