

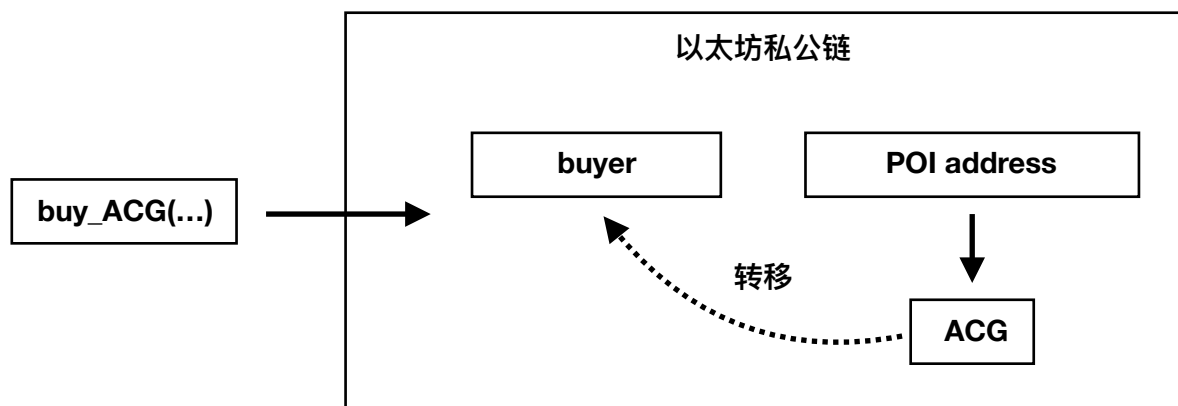
erc-20, erc-721 在艺术品交易中的使用

1. erc-20 股权币

- 已经部署在以太坊公网
- 总量固定（35亿）
- 不用于购买艺术品的交易
- 价值浮动

网页端web3接口

底层区块链



2. erc-20 交易币

- 将部署在我们的以太坊私有链上
- 总量不固定
- 价值与法币一样（美元/人民币）
- 用于购买艺术品的交易

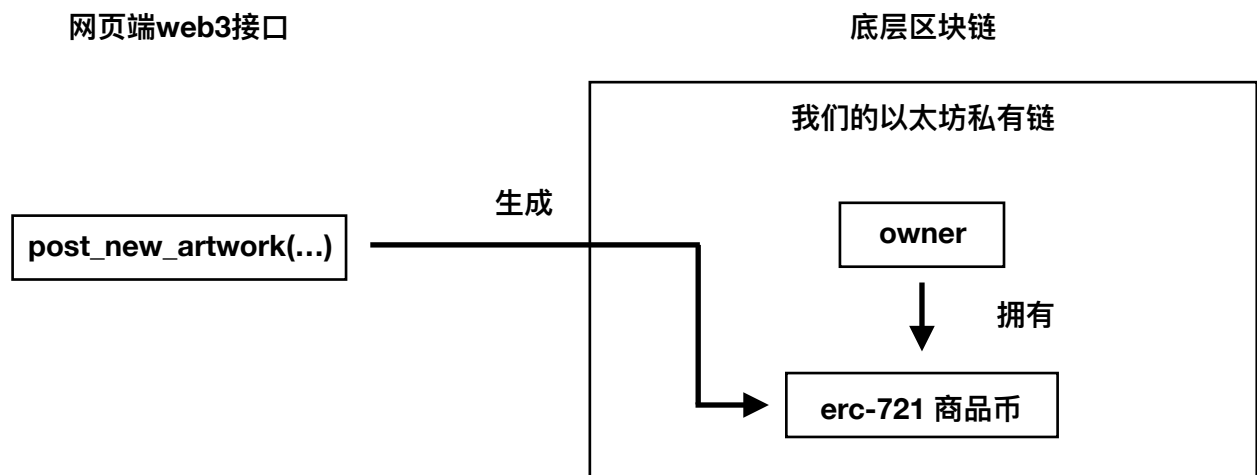
3. erc-721 商品币

- 将部署在我们的以太坊私有链上
- 总量不限
- 每个币 代表 一个独立存在的艺术品
- 每个币 包含 以下信息：id, name, owner, artist, price, type, size, material, etc...

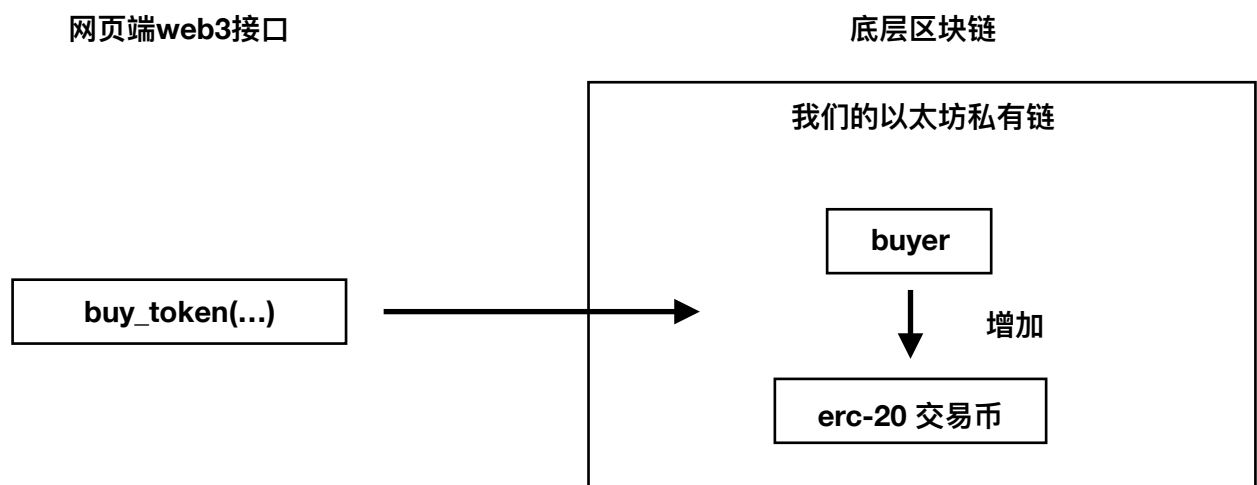
交易流程(不设计前端网站的审查判断机制)

除了第一个的 erc-20 股权币，erc-20 交易币 和 erc-721 商品币 将被用于 ArtChain Global 的艺术品交易，交易的基本流程如下：

1. 卖家**创建商品**，会产生一个新的 **erc-721** 商品币，记录着关于这件艺术品的基本信息

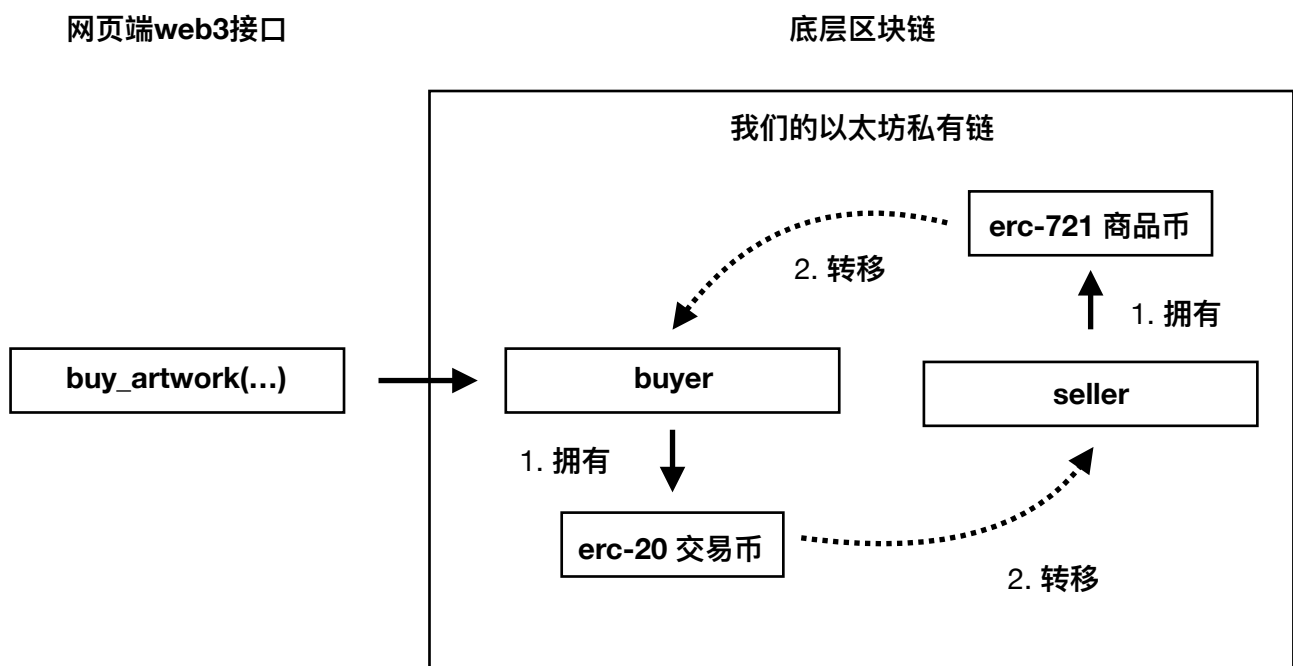


2. 买家（使用法币）**购买** erc-20 交易币



3. **购买艺术品**(也就是购买erc-721币)方式有两种
- 直接购买：支付足够数量的 erc-20 交易币，立即购买。
 - 参加拍卖：冻结足够数量的 erc-20 交易币，参加拍卖。
4. 如果购买成功，该商品对应的 erc-721 商品币中的 owner 讲改写成 买家

e.g.交易例子



POI模式/流程

任意新的交易被添加到区块链之前，必须有至少一位 **超级节点**（或者多位）确认。流程如下：

1. 所有超级节点接收到新的交易
2. 一个（或多个）超级节点确认
3. 普通节点同步添加新的区块
4. 即交易被确认

超级节点配置表

分类	配置	备注
CPU	16核 主频2.0GHZ以上	
内存	32G	
硬盘	16T 7200转SAS接口硬盘/16T SSD	
网络	双向带宽最低20Mbps 推荐双向带宽100Mbps	只有双向带宽满足100Mbps的候选节点，才可能成为高性能节点
系统	Ubuntu 16.04 64bit Centos 7.0及以上	

前段网站 和 底层私有链 联系所需要的 API

需要改写 链上数据的 操作 token = erc-20 交易币

// 建立新用户

```
1. add_new_user(user_address, new_user_prize){  
    // 需要做的事  
    1.1 将该 user_address 存到链上  
    1.2 该 user_address 的余额 = new_user_prize (这个prize 是 管理员后台可以设置的)  
}
```

// 发布新艺术品

```
2. post_new_artwork(owner, artwork_type, artist, loyalty, status, new_post_prize){  
    // 需要做的事  
    2.1 该 artwork owner = owner_address  
    2.2 该 artwork artist = artist_address  
    2.3 该 artwork loyalty = loyalty (loyalty 是管理员后台设置的吗?)  
    2.4 该 artwork 状态, 卖? 展示? 只存?  
    2.5 该 owner_address 余额 + new_post_prize (发布新艺术品的奖励, 后台可设置)  
  
    最后 返回一个 artwork_id  
}
```

// 购买艺术品

```
3. buy_artwork(buyer_address, owner_address, artwork_id, price){  
    // 要进行的操作  
    3.1 buyer_address 余额 - price  
    3.2 owner_address 余额 + price * (1-loyalty)  
    3.3 artist_address 余额 + price * loyalty  
    3.4 该 artwork owner = buyer_address  
  
    最后返回一个 transaction_id  
}
```

// 冻结token (用于拍卖行为)

```
4. freeze_token(buyer_address, artwork_id, price){  
    // 要进行的操作  
    4.1 判断 buyer_address 余额 大于 该艺术品的price  
    4.2 buyer_address 中的token冻结  
}
```

```

// 购买我们的交易币 （用于购买艺术品或拍卖）
5. buy_token(buyer_address, amount){

    // 这个接口需要和法币那边的接口对接,
    // 当法币账户成功收到钱了

    // 要进行的操作
    5.1 buyer_address 中的token增加 amount 个
}

```

仅仅查看链上数据的操作

```

// 查看某个艺术品
1. check_artwork( artwork_id ) {
    //返回的信息
    1.1 owner （现拥有者）
    1.2 artist （艺术家）
    1.3 status （卖？ 拍卖？ 只展示？ ）
    1.4 price （价格）
}

// 查看某个注册用户
2. check_user( user_id ) {
    //返回的信息
    1.1 title （画廊？ 艺术家？ 普通用户？ ）
    1.2 token_balance （该地址下的token数量）
    1.3 artworks （该地址下的 所有艺术品， 包括买到的， 要卖的， etc...）
}

// 查看某个交易记录
3. check_transaction( transaction_id ) {
    //返回的信息
    1.1 artwork_id (艺术品)
    1.2 artist （艺术家）
    1.3 buyer （买家）
    1.4 seller （卖家）
    1.5 price （价格）
    1.6 loyalty_rate
}

```