

# 超算链接口规范

版本记录

版本号	修改人	审批人	修改日期	修改记录
0.0.1	Ailen		2017/7/17	创建
0.0.2	Ailen		2017/11/2	修正
0.0.3	Jack		2018/1/20	修正

# 1 引言

## 1.1 编写目的

本文档为超算链进行交互的接口说明，作为用户自有系统平台与超算链的数据对接接口的标准与规范，指导开发人员进行接口设计、开发和测试。

## 1.2 超算链概述

超算链是一个完全独立开发、高效的区块链技术实现，是 ZeusNet 的核心引擎，ZeusNet 的灵魂。ZeusNet 是算力资源高效管理和调度的系统，需要完成全球算力资源交易的快速匹配撮合；通过准确调度和快速分发，将Dapp 运行环境下发到指定的算力资源设备上；为全球不同地方正在发生的算力资源租用交易提供公平公正的工作量统计、并完成交易的结算等，这些必须要依赖于一个强大、高效的区块链计算实现作为核心支撑。目前市面上现存的区块链 3.0 产品无法满足需求。超算链是区块链 4.0 的技术实现，在性能、可扩展性、安全性、稳定性等方面将会大大领先于区块链 3.0 技术。

超算链是在 Token 价值体系下，满足高性能交易，多用途计算能力的区块链平台。超算链是 1+9 的高性能交易主链，同时通过多条生态链实现不同资源的算力去中心化。

超算链核心优势包括：

1) 1+9 多链技术：1 条交易主链，实现各类服务和链的注册、登记、查询，计算撮合由另外的 9 条工作内链完成，通过马尔可夫链的随机相关性分析，实现不同交易根据时序快速成块和确认。

2) DDN (Docker Deliver Network)：容器分发网络。算力交易市场，关键问题在于在资源租用交易达成后，如何将需要相关算力使用的Dapps 运行环境分发到指定的算力资源设备上，正式开始算力资源的消耗，并在公平和高效之间寻找平衡点。超算链在架构设计过程中充分发挥服务分发路径优化，云计算，以及 Docker 容器分发等技术特点，实现了公平和高效间的平衡。

3) 代币发行机制：ZeusNet 将 ZNC 作为主币，可以分叉出不同的子币 (Token)，其它 Token 也可以锚定 ZNC 实现与其它数字货币 BTC/ETH 的兑换。例如 CDN 加速应用场景中，发行 CDNTOKEN，GPU 算力中发行 GToken，不同的游戏可以发行不同的 Token (德州、乐透等等) 等。

代币发行让不同的Dapps 系统内轻松实现独立的计费模型。每个 Token 和 ZNC 之间根据押金进行锚定，建立完善安全的价值体系。

4) 完整的生态结构：算力市场里面，需要建立存储、状态流转、接入前置等多个开发生态，才能让整个算力市场运作起来。

## 1.3 预期读者

Dapp 开发人员，测试人员。

## 1.4 术语与缩略语

资产 (asset)：一切放入区块链中的数据，统称为资产。

资产回溯：从区块链中将特定资产的历史变化按照时间顺序提取出来。

账本 (Ledger)：按照时间顺序将数据组成区块串连在一起，并保证数据不可篡改和伪造的分布式账本。

# 2 区块链监控平台

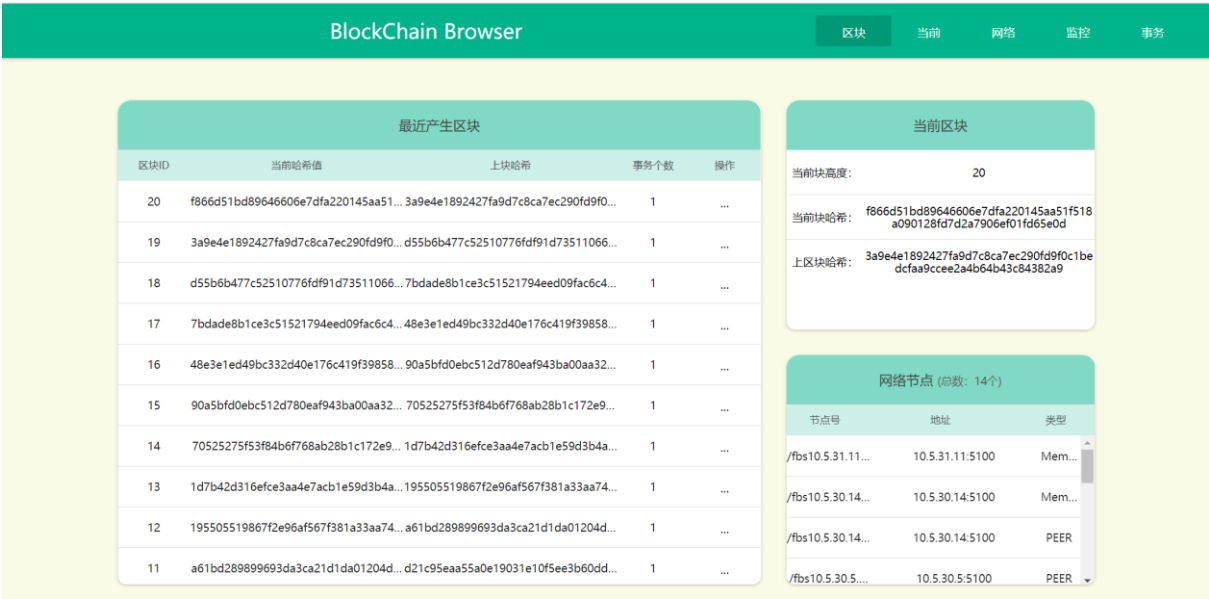
## 2.1 描述

区块链监控平台是用于监控链上数据的所有操作，和了解当前区块链的运行情况。如：监控最近产生区块、当前区块、网络节点数、当前算力、平均事务延迟、区块事务个数、最近提交的事务。

## 2.2 地址

<http://bc.zeusnet.io/prototype/browser/>

## 2.3 页面截图



## 3 测试账号

用于超算链功能的测试账号：账号名：test，密码：123456。

## 4 钱包接口说明

用户通过钱包开发调用接口，将自有系统和区块链间的系统对接。通过调用接口，实现资产、数据信息的入链、查询、溯源等区块链操作。超算链提供 http 形式的交互接口，包括账户登录接口、资产创建接口、文件上传接口、资产信息校验接口、资产交易接口以及交易查询接口，参数采用 post 方式提交 Json 格式报文。

## 4.1 账户登录

账本中的参与者账户进行登录接口，用于身份登录，获取钱包 IP。

### 4.1.1 请求报文

参数名	参数名称	类型及长度	备注	是否可空
orderNo	订单号	String(32)	每次请求的唯一编号	N
userName	用户名	String(64)		N
passwd	密码	String(64)		N
ledgerId	账本 ID	String(32)		N

例如：

```
{  
  "userName": "jack1",  
  "passwd": "1223",  
  "orderNo": "10011",  
  "ledgerId": "123"  
}
```

### 4.1.2 响应报文

参数名	参数名称	类型及长度	备注	是否可空
orderNo	订单号	String(32)	每次请求的唯一编号,返回与请求时相同的值	N
errorCode	返回的状态信息	String(6)	errorCode=000000（六个零）为成功标志，其他为失败	N
errorDesc	返回的信息	String(256)		N
userId	用户 ID	String(64)		N
userKey	用户公钥	String(128)		N

例如：

```
{  
  "errorCode": "000000",  
  "errorDesc": "success",  
  "userId": "0001110101001",  
  "userKey":  
    "04005f56800c1c8a2961979f033cfa3507f72a44b25ea6efb25a6392485e55c5b3214ce37437  
    418e397d723ce dc52cb219524f2383b45010a3e5aab9ea9304c7fc",  
  "orderNo": "10006"  
}
```

## 4.2 创建资产

客户可以通过创建资产将数据放入区块链中（即资产入链）。

创建资产需要创建资产别名、资产关键字、资产附属信息以及要放入区块链的数据。

### 4.2.1 请求报文

参数名	参数名称	类型及长度	备注	是否可空
orderNo	订单号	String(32)	每次请求的唯一编号	N
userKey	公钥 hash	String(64)		N
userId	用户 ID	String(64)		N
alias	资产别名	String(32)		N
dataTable	附加数据检索词	String(50)	以^分割如: def^book	N
filePath	文件存放路径	String (160)	以^分割	N
metadata	附加数据	String(1548)		N
type	资产类型	String(2)	资产类型 (0=Token , 1=FBC, 2=BTC, 3=LTC)	N

例如:

```
{  
  "orderNo": "10013",  
  "userKey": "11",  
  "userId": "0001110101001",  
  "alias": "def",  
  "dataTable": "123",  
  "filePath": "11",  
  "type": "1",  
  "metadata": "1"  
}
```

### 4.2.2 响应报文

参数名	参数名称	类型及长度	备注	是否可空
orderNo	订单号	String(32)	每次请求的唯一编号, 返回与请求时相同的值	N
errorCode	返回的状态信息	String(6)	状态码, 000000 为成功	N
errorDesc	返回的信息	String(256)	成功, 则为空, 否则为返回描述	N
assetId	资产 ID	String(64)	成功, 返回资产 ID	N

例如：

```
{
  "orderNo": "10013",
  "errorCode": "000000",
  "errorDesc": "success",
  "assetId": "ff808081000065c5015d7cf740460004"
}
```

## 4.3 文件上传

用户接口，用于用户上传数据。文件上传是通过 form 表单的形式进行上传的。需要在 URL 后面跟上下面这些参数。

### 4.3.1 请求报文

参数名	参数名称	类型及长度	备注	是否可空
orderId	订单号	String(32)		N
userId	用户 ID	String(64)		N
userKey	用户公钥	String(128)		N

例如：

orderId=10039& userId=0001110101001& userKey=123

### 4.3.2 响应报文

参数名	参数名称	类型及长度	备注	是否可空
orderId	订单号	String(32)		N
errorCode	返回状态码	String(6)		N
errorDesc	返回状态信息	String(256)		N
fileId	文件 id	String(64)	以^分割	N

例如：

```
{
  "fileId": "[ff80808100006f43015d7d8e0f500004]",
  "orderId": "10039",
  "errorDesc": "success",
  "errorCode": "000000"
}
```



## 4.4 文件下载

用户接口，用于用户下载数据。需要在 URL 后面跟上下面这些参数。

### 4.4.1 请求报文

参数名	参数名称	类型及长度	备注	是否可空
orderId	订单号	String(32)		N
userId	用户 ID	String(64)		N
fileId	文件主键	String(64)		N

例如：

`fileId=ff8080810000705d015d7da921260001&orderId=10057&userId=0001110101001`

### 4.4.2 响应报文

参数名	参数名称	类型及长度	备注	是否可空
errorCode	返回状态码	String(6)	失败时不为空	N
errorDesc	返回状态信息	String(256)	失败时不为空	N

例如：

```
{  
  "errorDesc": "000111010100133: 用户不存在",  
  "errorCode": "000001"  
}
```

## 4.5 资产查询

对用户放入区块链上的数据进行查询。

### 4.5.1 请求报文

参数名	参数名称	类型及长度	备注	是否可空
orderNo	订单号	String(32)	每次请求的唯一编号	N
userId	用户 ID	String(64)		N
userKey	用户公钥	String(128)	登录时获得的公钥	N
dataTable	附加数据检索词	String(50)	以^分割	N
pageNo	页码	Int		N
pageSize	页容量	Int		N

例如：

```
{
  "orderNo": "2000001",
  "userId": "xxxxxxxxxx",
  "userKey": "xxxxxxxxxx",
  "dataTable": "金融^票据",
  "pageNo": 1,
  "pageSzie": 20
}
```

## 4.5.2 响应报文

参数名	参数名称	类型及长度	备注	是否可空
orderNo	订单号	String(32)	每次请求的唯一编号,返回与请求时相同的值	N
errorCode	返回状态码	String(6)		N
errorDesc	返回状态信息	String(256)		N
assetCount	资产总数量	Int		N
pageNo	页码	Int		N
assets	资产列表	List<Asset>		N

例如：

```
{
  "orderNo": "10021",
  "errorCode": "000000",
  "errorDesc": "success",
  "assetCount": 3,
  "pageNo": 0,
  "assets": [
    {
      "assetId": "ff80808100006de4015d7d2b3a470001",
      "type": "1",
      "alias": "def",
      "dataTable": "123",
      "metadata": "1",
      "amount": 0,
      "count": 0,
      "status": "1",
      "createTime": "2017-07-26 12:32:52",
      "updateTime": "2017-07-26 12:32:52"
    }
  ],
}
```

```

{
  "assetId": "ff808081000065c5015d7cf740460004",
  "type": "1",
  "alias": "def",
  "dataTable": "123",
  "metadata": "1",
  "filePath": "11",
  "amount": 0,
  "count": 0,
  "status": "1",
  "createTime": "2017-07-26 11:36:05",
  "updateTime": "2017-07-26 11:36:05"
},
{
  "assetId": "ff80808100006128015d7cb9ed1a0002",
  "type": "0",
  "alias": "default",
  "amount": 0,
  "count": 0,
  "status": "1",
  "createTime": "2017-07-26 10:29:06",
  "updateTime": "2017-07-26 10:29:06"
}
]
}

```

## 4.6 资产回溯

对区块链上操作后的数据进行向上推导。当有故障发生时，可以快速有效的定位、从而减少故障时间；当网络受到攻击时，可以快速定位、分析、取证。

### 4.6.1 请求报文

参数名	参数名称	类型及长度	备注	是否可空
orderId	订单号	String(32)	每次请求的唯一编号	N
userId	用户 id	String(64)		N
userKey	用户公钥	String(128)		N
assetId	资产 ID	String(64)		N
level	回溯层数	Int	回溯层数，最大为 5	N

例如：

```
{  
  "orderId": "10025",  
  "userKey": "11",  
  "userId": "0001110101001",  
  "assetId": "ff80808100006de4015d7d2b3a470001",  
  "level": "1"  
}
```

### 4.6.2 响应报文

参数名	参数名称	类型及长度	备注	是否可空
orderId	订单号	String(32)	每次请求的唯一编号，返回与请求时相同的值	N
errorCode	返回状态码	String(6)		N
errorDesc	返回状态信息	String(256)		N
transactions	交易	List<Transaction>		N

例如：

```
{
  "orderId": "10025",
  "errorCode": "000000",
  "errorDesc": "success",
  "transactions": [
    {
      "assetId": "ff80808100006de4015d7d2b3a470001",
      "type": "1",
      "alias": "def",
      "dataTable": "123",
      "metadata": "1",
      "amount": 0,
      "count": 0,
      "status": "1",
      "createTime": "2017-07-26 12:32:52",
      "updateTime": "2017-07-26 12:32:52"
    }
  ]
}
```

## 5 附录

### 5.1 接口类型

序号	状态名称	状态值
1	基础类接口	01
2	交易类接口	02
3	查询类接口	03

## 5.2 接口名称

接口名称	模块名称	接口编码	接口类型
用户登录	user	002	01
创建资产	ast	005	02
文件上传	sys	006	03
文件下载	sys	007	04
资产查询	ast	008	05
资产回溯	ast	009	06

## 5.3 接口地址

接口名称	接口地址
用户登录	/usr/pblin.do?fh=LINUSR0000000J00&resp=bd&bd
新建账本	/ldg/pbnew.do?fh=NEWLDG0000000J00&resp=bd&bd
创建资产	/ast/pbcrt.do?fh=CRTAST0000000J00&resp=bd&bd
文件上传	/sys/upload.do
文件下载	/sys/download.do
资产查询	/ast/pbqry.do?fh=QRYAST0000000J00&resp=bd&bd
资产回溯	/ast/pbtra.do?fh=TRAAST0000000J00&resp=bd

## 5.4 资产 (ASSET) 字段说明

名称	说明	类型及长度	备注
assetId	资产 ID	String(64)	
type	资产类型	String(8)	FBC; BTC; LTC; TOKEN
alias	别名	String(32)	
dataTable	查询关键字	String(50)	
metadata	附属信息	String(1548)	
filePath	附件 id	String(50)	用^分隔
amount	金额	Decimal(10,4)	
count	数量	Decimal(10,0)	
status	状态	String(12)	0=无效; 1=创建; 2=入链; 3=有效
creatTime	创建时间	datetime	
updateTime	更新时间	datetime	

## 5.5 交易 (Transaction) 字段说明

名称	说明	类型及长度	备注
transactionId	交易 ID	String(64)	
srcAssetId	源资产 ID	String(8)	
fromUserId	发起方用户 ID	String(32)	
fromUserName	发起方用户名	String(50)	
destAssetId	目标资产 ID	String(1548)	
toUserId	目标用户 ID	String(50)	
toUserName	目标用户名	String(50)	
amount	金额	Decimal(10,4)	
count	数量	Decimal(10,0)	
status	状态	String(10)	0=无效; 1:创建; 2: 入链; 3: 有效
createTime	创建时间	datetime	
endTime	截止时间	datetime	