BDWare Documentation

Search docs

Contents

HTTP APIs

Contents

go-bdledger's HTTP APIs are generated from gRPC APIs using grpc-gateway, with the following config:

```
type: google.api.Service
config_version: 3

http:
  rules:
    - selector: bdware.bdledger.api.Node.ClientVersion
      get: /v0/node/version
    - selector: bdware.bdledger.api.Ledger.CreateLedger
      post: /v0/ledgers
      body: "*"
    - selector: bdware.bdledger.api.Ledger.GetLedgers
```

```
      get: /v0/ledgers
    - selector: bdware.bdledger.api.Ledger.SendTransaction
      post: /v0/ledgers/{ledger}/transactions
      body: "*"
    - selector: bdware.bdledger.api.Query.GetBlockByHash
      get: /v0/ledgers/{ledger}/block
    - selector: bdware.bdledger.api.Query.GetBlocks
      post: /v0/ledgers/{ledger}/blocks/query
      body: "*"
    - selector: bdware.bdledger.api.Query.CountBlocks
      post: /v0/ledgers/{ledger}/blocks/count
      body: "*"
    - selector: bdware.bdledger.api.Query.GetRecentBlocks
      get: /v0/ledgers/{ledger}/blocks/recent
    - selector: bdware.bdledger.api.Query.GetTransactionByHash
      get: /v0/ledgers/{ledger}/transaction
    - selector: bdware.bdledger.api.Query.GetTransactionByBlockHashAndIndex
      get: /v0/ledgers/{ledger}/block/transaction
    - selector: bdware.bdledger.api.Query.GetTransactions
      post: /v0/ledgers/{ledger}/transactions/query
      body: "*"
    - selector: bdware.bdledger.api.Query.CountTransactions
      post: /v0/ledgers/{ledger}/transactions/count
      body: "*"
```

NOTE

Request/Response data of **bytes** type should/will be encoded with Base64.

NOTE

When using hash strings in URL, they need to be encoded with encodeURIComponent.

1. Request Examples

1.1. Node.ClientVersion

Get BDLedger node version

```
GET http://{{IP}}:{{PORT}}/v0/node/version
```

Response

```
{
  "version": "dev-210119.a88bf4eb"
}
```

1.2. Ledger.CreateLedger

Create a new ledger

```
POST http://{{IP}}:{{PORT}}/v0/ledgers
```

Request body

```
{
  "name": "test"
}
```

Response

```
{
  "ok": true
}
```

1.3. Ledger.GetLedgers

Get all ledgers

```
GET http://{{IP}}:{{PORT}}/v0/ledgers
```

Response

```
{
  "ledgers": [
    "default",
    "test"
  ]
}
```

1.4. Ledger.SendTransaction

Send a new transaction

```
POST http://{{IP}}:{{PORT}}/v0/ledgers/test/transactions
```

Request body

```
{
  "transaction": {
    "type": 0,
    "from": "8A3K/vANyv7wDcr+8A3K/vANyv4=",
    "nonce": 52,
    "data": "lQItWZKS5hlUn6V/DMKKwvZXxvM="
  }
}
```

Response

```
{
  "hash": "VQBeA5Ee0Y5hqEileoQuYMHbOSE="
```

```
}
```

1.5. Query.GetBlockByHash

Get a block identified by its hash

```
GET http://{{IP}}:{{PORT}}/v0/ledgers/test/block?
hash=LSKr%2BK079Ax%2BrKdlyYN5ze2YGzo%3D
```

hash has to be encoded with encodeURIComponent

Response

```
{
  "block": {
    "hash": "LSKr+K079Ax+rKdlyYN5ze2YGzo=",
    "creator": "",
    "nonce": "0",
    "parentHashes": [
      "fLX5pMY8M1qSAGZdKT1rWBkdEMo=",
      "rk0DWMaUpRG82yVX+cFhbfhPFdw=",
      "3XkwkuMBearq8uavN76Te7Zdpl8="
    ],
    "witnesses": [],
    "timestamp": "1611038043",
    "size": "0",
    "transactionCount": 1,
    "transactionsRoot": "VQBeA5Ee0Y5hqEileoQuYMHbOSE=",
    "transactions": [
      {
        "blockHash": "",
        "blockTimestamp": "0",
        "index": 0,
        "hash": "VQBeA5Ee0Y5hqEileoQuYMHbOSE=",
        "type": "RECORD",
        "from": "8A3K/vANyv7wDcr+8A3K/vANyv4=",
        "nonce": "0",
        "to": "AAAAAAAAAAAAAAAAAAAAAAAAAAA=",
        "data": "lQItWZKS5hlUn6V/DMKKwvZXxvM="
      }
    ],
    "transactionHashes": [
      "VQBeA5Ee0Y5hqEileoQuYMHbOSE="
    ]
  }
}
```

1.6. Query.GetBlocks

Get blocks in a timestamp range

```
POST http://{{IP}}:{{PORT}}/v0/ledgers/test/blocks/query
```

```
enum IncludeTransactions {
  NONE = 0; // Don't include transaction data
  HASH = 1; // Include transactions hashes
  FULL = 2; // Include full transactions
}
```

Requirement: $\text{start\_timestamp} \leq \text{end\_timestamp}$

If neither **start_timestamp** nor **end_timestamp** is specified, then **start_timestamp** will be set to the genesis block's timestamp, and **end_timestamp** will be set to $\text{start\_timestamp} + \text{query.maxDuration}$ (**query.maxDuration** is specified in go-bdledger's config file).

If only **end_timestamp** is not specified, or $\text{end\_timestamp} - \text{start\_timestamp} > \text{query.maxDuration}$, then **end_timestamp** will be set to $\text{start\_timestamp} + \text{query.maxDuration}$.

If only **start_timestamp** is not specified, then **start_timestamp** will be set to $\text{end\_timestamp} - \text{query.maxDuration}$.

In all cases, **start_timestamp** will never be earlier than the genesis block's timestamp, and **end_timestamp** will never be later than the current timestamp when the node process the query request.

Request body 1

```
{
  "start_timestamp": 1611038000,
  "end_timestamp": 1611039000,
  "include_transactions": 0
}
```

Response 1

```
{
  "blocks": [
    {
      "hash": "LSKr+K079Ax+rKdlyYN5ze2YGzo=",
      "creator": "",
      "nonce": "0",
      "parentHashes": [
        "fLX5pMY8M1qSAGZdKT1rWBkdEMo=",
        "rk0DWMaUpRG82yVX+cFhbfhPFdw=",
        "3XkwkuMBearq8uavN76Te7Zdpl8="
      ],
      "witnesses": [],
      "timestamp": "1611038043",
```

```
      "size": "0",
      "transactionCount": 1,
      "transactionsRoot": "VQBeA5Ee0Y5hqEileoQuYMHbOSE=",
      "transactions": [],
      "transactionHashes": []
    }
  ],
  "startTimestamp": "1611038043",
  "endTimestamp": "1611038043"
}
```

Request body 2

```
{
  "start_timestamp": 1611038000,
  "end_timestamp": 1611039000,
  "include_transactions": 1
}
```

Response 2

```
{
  "blocks": [
    {
      "hash": "LSKr+K079Ax+rKdlyYN5ze2YGzo=",
      "creator": "",
      "nonce": "0",
      "parentHashes": [
        "fLX5pMY8M1qSAGZdKT1rWBkdEMo=",
        "rk0DWMaUpRG82yVX+cFhbfhPFdw=",
        "3XkwkuMBearq8uavN76Te7Zdpl8="
      ],
      "witnesses": [],
      "timestamp": "1611038043",
      "size": "0",
      "transactionCount": 1,
      "transactionsRoot": "VQBeA5Ee0Y5hqEileoQuYMHbOSE=",
      "transactions": [],
      "transactionHashes": [
        "VQBeA5Ee0Y5hqEileoQuYMHbOSE="
      ]
    }
  ],
  "startTimestamp": "1611038043",
  "endTimestamp": "1611038043"
}
```

Request body 3

```
{
  "start_timestamp": 1611038000,
  "end_timestamp": 1611039000,
```

```
  "include_transactions": 2
}
```

Response 3


```
{
  "blocks": [
    {
      "hash": "LSKr+K079Ax+rKdlyYN5ze2YGzo=",
      "creator": "",
      "nonce": "0",
      "parentHashes": [
        "fLX5pMY8M1qSAGZdKT1rWBkdEMo=",
        "rk0DWMaUpRG82yVX+cFhbfhPFdw=",
        "3XkwkuMBearq8uavN76Te7Zdpl8="
      ],
      "witnesses": [],
      "timestamp": "1611038043",
      "size": "0",
      "transactionCount": 1,
      "transactionsRoot": "VQBeA5Ee0Y5hqEileoQuYMHbOSE=",
      "transactions": [
        {
          "blockHash": "",
          "blockTimestamp": "0",
          "index": 0,
          "hash": "VQBeA5Ee0Y5hqEileoQuYMHbOSE=",
          "type": "RECORD",
          "from": "8A3K/vANyv7wDcr+8A3K/vANyv4=",
          "nonce": "0",
          "to": "AAAAAAAAAAAAAAAAAAAAAAAAAAA=",
          "data": "lQItWZKS5hlUn6V/DMKKwvZXxvM="
        }
      ],
      "transactionHashes": [
        "VQBeA5Ee0Y5hqEileoQuYMHbOSE="
      ]
    }
  ],
  "startTimestamp": "1611038043",
  "endTimestamp": "1611038043"
}
```


1.7. Query.CountBlocks

Count all blocks in a ledger, or blocks in a timestamp range

```
POST http://{{IP}}:{{PORT}}/v0/ledgers/test/blocks/count
```

Requirement: $\text{start\_timestamp} \leq \text{end\_timestamp}$

If neither **start_timestamp** nor **end_timestamp** is specified, then count all blocks in the specified ledger.

If only **end_timestamp** is not specified, then count all blocks with timestamps later than **start_timestamp**.

If only **start_timestamp** is not specified, then count all blocks with timestamps earlier than **end_timestamp**.

In all cases, **start_timestamp** will never be earlier than the genesis block's timestamp, and **end_timestamp** will never be later than the current timestamp when the node process the query request.

Request body 1

```
{}
```

Response 1

```
{
  "count": "5",
  "startTimestamp": "0",
  "endTimestamp": "1611039957"
}
```

Request body 2

```
{
  "start_timestamp": 1611038000,
  "end_timestamp": 1611039000
}
```

Response 2

```
{
  "count": "1",
  "startTimestamp": "1611038000",
  "endTimestamp": "1611039000"
}
```

1.8. Query.GetRecentBlocks

Get recent **count** blocks (Only support IncludeTransactions=NONE for now)

```
GET http://{{IP}}:{{PORT}}/v0/ledgers/test/blocks/recent?count=2
```

Response

```
{
  "blocks": [
    {
```

```
      "hash": "LSKr+K079Ax+rKdlyYN5ze2YGzo=",
      "creator": "",
      "nonce": "0",
      "parentHashes": [
        "fLX5pMY8M1qSAGZdKT1rWBkdEMo=",
        "rk0DWMaUpRG82yVX+cFhbfhPFdw=",
        "3XkwkuMBearq8uavN76Te7Zdpl8="
      ],
      "witnesses": [],
      "timestamp": "1611038043",
      "size": "0",
      "transactionCount": 1,
      "transactionsRoot": "VQBeA5Ee0Y5hqEileoQuYMHbOSE=",
      "transactions": [],
      "transactionHashes": []
    },
    {
      "hash": "rk0DWMaUpRG82yVX+cFhbfhPFdw=",
      "creator": "",
      "nonce": "0",
      "parentHashes": [
        "fLX5pMY8M1qSAGZdKT1rWBkdEMo=",
        "3XkwkuMBearq8uavN76Te7Zdpl8=",
        "8pZPR74OALIbps5XFb4dL/s0j0M="
      ],
      "witnesses": [],
      "timestamp": "1610968019",
      "size": "0",
      "transactionCount": 1,
      "transactionsRoot": "LuxttCm/pSHVMOKF0sJExk+DJXc=",
      "transactions": [],
      "transactionHashes": []
    }
  ],
  "startTimestamp": "1610968019",
  "endTimestamp": "1611038043"
}
```


1.9. Query.GetTransactionByHash

Get a transaction identified by its hash

```
GET http://{{IP}}:{{PORT}}/v0/ledgers/test/transaction?
hash=VQBeA5Ee0Y5hqEileoQuYMHbOSE%3D
```

hash has to be encoded with encodeURIComponent

Response


```
{
  "transaction": {
    "blockHash": "LSKr+K079Ax+rKdlyYN5ze2YGzo=",
```

```
    "blockTimestamp": "1611038043",
    "index": 0,
    "hash": "VQBeA5Ee0Y5hqEileoQuYMHbOSE=",
    "type": "RECORD",
    "from": "8A3K/vANyv7wDcr+8A3K/vANyv4=",
    "nonce": "0",
    "to": "AAAAAAAAAAAAAAAAAAAAAAAAAAA=",
    "data": "lQItWZKS5hlUn6V/DMKKwvZXxvM="
  }
}
```

1.10. Query.GetTransactionByBlockHashAndIndex

Get une transaction identified by hash of the block it belongs to and its index inside the block

```
GET http://{{IP}}:{{PORT}}/v0/ledgers/test/block/transaction?
blockHash=LSKr%2BK079Ax%2BrKdlyYN5ze2YGzo%3D&index=0
```

blockHash has to be encoded with encodeURIComponent

Response

```
{
  "transaction": {
    "blockHash": "LSKr+K079Ax+rKdlyYN5ze2YGzo=",
    "blockTimestamp": "1611038043",
    "index": 0,
    "hash": "VQBeA5Ee0Y5hqEileoQuYMHbOSE=",
    "type": "RECORD",
    "from": "8A3K/vANyv7wDcr+8A3K/vANyv4=",
    "nonce": "0",
    "to": "AAAAAAAAAAAAAAAAAAAAAAAAAAA=",
    "data": "lQItWZKS5hlUn6V/DMKKwvZXxvM="
  }
}
```

1.11. Query.GetTransactions

Get transactions in a timestamp range

```
POST http://{{IP}}:{{PORT}}/v0/ledgers/test/transactions/query
```

start_timestamp and **end_timestamp** follow the same requirements and rules as in Query.GetBlocks.

Request body

```
{
  "start_timestamp": 1611038000,
  "end_timestamp": 1611039000
}
```

Response


```
{
  "transactions": [
    {
      "blockHash": "",
      "blockTimestamp": "0",
      "index": 0,
      "hash": "VQBeA5Ee0Y5hqEileoQuYMHbOSE=",
      "type": "RECORD",
      "from": "8A3K/vANyv7wDcr+8A3K/vANyv4=",
      "nonce": "0",
      "to": "AAAAAAAAAAAAAAAAAAAAAAAAAAA=",
      "data": "lQItWZKS5hlUn6V/DMKKwvZXxvM="
    }
  ],
  "startTimestamp": "1611038043",
  "endTimestamp": "1611038043"
}
```


1.12. Query.CountTransactions

Count all transactions in a ledger, or transactions in a timestamp range

```
POST http://{{IP}}:{{PORT}}/v0/ledgers/test/transactions/count
```

start_timestamp and **end_timestamp** follow the same requirements and rules as in Query.CountBlocks.

Request body 1

```
{}
```

Response 1


```
{
  "count": "4",
  "startTimestamp": "0",
  "endTimestamp": "1611039957"
}
```


Request body 2

```
{
  "start_timestamp": 1611038000,
  "end_timestamp": 1611039000
}
```

Response 2


```
{
  "count": "1",
```

```
  "startTimestamp": "1611038000",
  "endTimestamp": "1611039000"
}
```

© The BDWare Authors 2020 | Documentation Distributed under CC-BY-4.0