
Expie API Documentation

Release 0.0.1

expie

Feb 06, 2018

Contents:

| | | |
|----------|-------------------------|----------|
| 1 | HTTP-API | 1 |
| 1.1 | General Usage | 1 |
| 1.2 | Order API | 2 |
| 1.3 | Markets API | 5 |
| 1.4 | Balance API | 7 |
| 2 | WebSocket API | 9 |
| 2.1 | General Usage | 9 |
| 2.2 | System API | 10 |
| 2.3 | Depth API | 10 |
| 2.4 | Price API | 12 |
| 2.5 | Deal API | 13 |

1.1 General Usage

How to use Expie API?

1.1.1 Access Method

Domain: **api.expie.com**

All API should be accessed with `auth_token` (in HTTP Authorization), and you can apply for your API `auth_token` in Bitpie APP.

1.1.2 Access Samples

Python:

```
domain = 'api.expie.com'
rootPath = '/v1'
url = 'https://' + domain + rootPath + '/markets'
token = 'e0b20185864172a0e209836e29d94be9fd2ca85c0a3ef4641f502d3c085820ea'
request = urllib2.Request(url)
request.add_header("Authorization", "Bearer" + token)
```

Curl:

```
curl -H "Authorization: Bearer_
↪e0b20185864172a0e209836e29d94be9fd2ca85c0a3ef4641f502d3c085820ea" https://api.expie.
↪com/v1/markets
```

1.2 Order API

1.2.1 Market Order Place

POST /markets/{market}/orders

Example response:

```
{
  "deal_fee": "0",
  "ctime": 1517801276.820693,
  "maker_fee": "0.0006",
  "price": "0.154",
  "deal_stock": "0",
  "side": 1,
  "source": "expie.api.https",
  "amount": "0.02",
  "user": 100056,
  "mtime": 1517801276.820693,
  "deal_money": "0",
  "left": "0.02",
  "type": 1,
  "id": 2169,
  "market": "BCCBTC",
  "taker_fee": "0.0006"
}
```

URL:

- market (*required*) - market name, for example *(BCCBTC).

Parameters:

- side (*required*) (*int*) - trade type, for example (1).
- amount (*required*) (*float*) - count or amount.
- price (*optional*) (*float*) - price.

Note:

- side 1: sell, 2: buy.
 - **amount count or amount.** At market price buy “amount” means spend your stock(usually”BTC”), at market price sell “amount” means the coin count you sold.
 - price price. If your URL body doesn’t have price params that means Market Order, else means Limit Order.
-

1.2.2 Market Order Cancel

POST /markets/{market}/orders/{orderId}/_cancel

Example url: /markets/BCCBTC/orders/2168/_cancel

Example response:

```
{
  "deal_fee": "0",
  "ctime": 1517799540.747482,
  "maker_fee": "0.0006",
  "price": "0.154",
  "deal_stock": "0",
  "side": 1,
  "source": "expie.api.https",
  "amount": "0.02",
  "user": 100056,
  "mtime": 1517799540.747482,
  "deal_money": "0",
  "left": "0.02",
  "type": 1,
  "id": 2168,
  "market": "BCCBTC",
  "taker_fee": "0.0006"
}
```

URL:

- market (*required*) - market name, for example (*BCCBTC*).
- orderId (*required*) - id, for example (*2168*).

1.2.3 Market Order Pending

GET /markets/{market}/orders/_pending

Example url: /markets/BTGBTC/orders/_pending?limit=10&offset=0

Example response:

```
{
  "records": [
    {
      "deal_fee": "0",
      "ctime": 1517580449.685034,
      "maker_fee": "0.001",
      "price": "0.001254",
      "deal_stock": "0",
      "side": 2, // 1: sell, 2: buy
      "source": "bitpie.client",
      "amount": "30",
      "user": 259285,
      "mtime": 1517580449.685034,
      "deal_money": "0",
      "left": "30",
      "type": 1, // 1: limit order, 2: market order
      "id": 20321,
      "market": "BTGBTC",
      "taker_fee": "0.001"
    }
  ],
  "total": 1,
  "limit": 10,
  "offset": 0
}
```

URL:

- market (*required*) - market name, for example **(BCCBTC)*.

Parameters:

- offset (*optional*) (*int*) - sinceId.
- limit (*optional*) (*int*) - limit.

1.2.4 Market Order Details

GET /markets/{market}/orders/{orderId}

Example url: /markets/BTGBTC/orders/13479

Example response:

```
{
  "records": [
    {
      "fee": "0.0010194",
      "deal": "0.0161197722",
      "price": "0.015813",
      "amount": "1.0194",
      "role": 1,
      "user": 259285,
      "time": 1517224387.037182,
      "deal_order_id": 13506,
      "id": 5288
    }
  ],
  "limit": 20,
  "offset": 0
}
```

URL:

- market (*required*) - market name, for example *(BTGBTC)*.
- orderId (*required*) - id,for example *(2168)*.

1.2.5 Market Order Finished

GET /markets/{market}/orders/_finished

Example url: /markets/BCCBTC/orders/_finished?limit=10&offset=0

Example response:

```
{
  "records": [
    {
      "ctime": 1516689377.805243,
      "maker_fee": "0",
      "price": "0.151",
      "deal_fee": "0",
      "id": 187,
      "source": "bitpie.client",
      "amount": "0.001",
    }
  ]
}
```



```

        "ftime": 1516689386.619962,
        "user": 116480,
        "deal_stock": "0.001",
        "deal_money": "0.000151",
        "type": 1,
        "side": 1,
        "market": "BCCBTC",
        "taker_fee": "0"
    }
],
"limit": 10,
"offset": 0
}

```

URL:

- market (*required*) - market name, for example **(BCCBTC)*.

Parameters:

- offset (*optional*) (*int*) - sinceId.
- limit (*optional*) (*int*) - limit.

1.3 Markets API

1.3.1 Markets

GET /markets

Example response:

```

[
  {
    "name": "BCCBTC",
    "market_order_min_money": "0.001",
    "stock_precision": 4,
    "money": "BTC",
    "maker_fee_rate": 0,
    "order_min_vol": "0.001",
    "money_precision": 6,
    "market_order_enabled": false,
    "stock": "BCC",
    "enabled": true,
    "taker_fee_rate": 0
  },
  {
    "name": "BTGBTC",
    "market_order_min_money": "0.001",
    "stock_precision": 4,
    "money": "BTC",
    "maker_fee_rate": 0.001,
    "order_min_vol": "0.001",
    "money_precision": 6,
    "market_order_enabled": false,
    "stock": "BTG",
    "enabled": true,
  }
]

```

```
    "taker_fee_rate": 0.001
  }
]
```

1.3.2 Markets Depth

GET /markets/{market}/depth

Example url: /markets/BCCBTC/depth

Example response:

```
{
  "bids": [
    ["0.133", "0.0069"],
    ["0.132", "1"],
    ["0.13", "1"],
    ["0.122508", "1.3"],
    ["0.122001", "1.2"],
    ["0.122", "1"],
    ["0.12", "0.6151"],
    ["0.111001", "5"],
    ["0.111", "3.1927"],
    ["0.1022", "50"]],
  "asks": [
    ["0.1363", "1.5"],
    ["0.139999", "1.5"],
    ["0.14", "3"],
    ["0.156", "5.9824"],
    ["0.16", "5.0397"],
    ["0.199999", "5"],
    ["0.2", "40.4095"],
    ["0.22", "1.7496"],
    ["0.25", "1.035"],
    ["0.33", "3.1"]
  ]
}
```

URL:

- market (*required*) - market name, for example **(BCCBTC)*.

1.3.3 Markets Ticker

GET /markets/{market}/ticker

Example url: /markets/BCCBTC/ticker

Example response:

```
{
  "high": "0.139999",
  "ask": "0.1363",
  "bid": "0.134",
  "last": "0.1345",
  "low": "0.1345"
}
```

URL:

- market (*required*) - market name, for example *(BCCBTC).

1.4 Balance API

1.4.1 Balance summary

GET /balance/summary**Example response:**

```
[
  {
    "coin_code": "BTC",
    "available_amount": "0",
    "balance": "0",
    "ex_trading_amount": "0",
    "ex_trading_amount_str": "0",
    "pending_deposit_amount": "0",
    "ex_available": "0",
    "escrow_amount": "0",
    "min_deposit": "0.001",
    "pending_withdraw_amount": "0",
    "pending_deposit_amount_str": "0",
    "pending_withdraw_amount_str": "0",
    "ex_freeze_str": "0",
    "min_withdraw": "0.001",
    "escrow_amount_str": "0",
    "ex_available_str": "0",
    "available_amount_str": "0",
    "ex_freeze": "0",
    "balance_str": "0"
  }
]
```

1.4.2 Balance history

GET /balance/history (*String: coin_code*) (*int: limit*) (*int: offset*)

Example url: /balance/history?coin_code=BCC&limit=10&offset=0

Example response:

```
{
  "records": [
    {
      "business": "deposit",
      "detail": {
        "timestamp_action_id": 1517749392,
        "id": 85130
      },
      "asset": "BCC",
      "time": 1517749392.541296,
      "balance": "1",
      "change": "0.9"
    }
  ]
}
```

```
    },
    {
      "business": "deposit",
      "detail": {
        "timestamp_action_id": 1517749197,
        "id": 85128
      },
      "asset": "BCC",
      "time": 1517749197.094249,
      "balance": "0.1",
      "change": "0.1"
    }
  ],
  "limit": 10,
  "offset": 0
}
```

Parameters:

- `coin_code` (*required*) (*String*) - coinCode.
- `offset` (*optional*) (*int*) - offset.
- `limit` (*optional*) (*int*) - limit.

2.1 General Usage

How to use Expie WebSocket?

2.1.1 Access Method

Link address for WebSocket: <wss://api.expie.com/ws>

2.1.2 Access Samples

WebSocket Client sends heartbeat:

```
{
  "method": "server.ping",
  "params": [],
  "id": 1223
}
```

Response:

```
{
  "error": null,
  "result": "pong",
  "id": 1223
}
```

2.2 System API

2.2.1 PING

Example request:

```
{
  "method": "server.ping",
  "params": [],
  "id": 1223
}
```

Response:

```
{
  "error": null,
  "result": "pong",
  "id": 1223
}
```

2.2.2 System Time

Example request:

```
{
  "method": "server.time",
  "params": [],
  "id": 1223
}
```

Response:

```
{
  "error": null,
  "result": 1517455018,
  "id": 1223
}
```

2.3 Depth API

2.3.1 Depth query

Example request:

```
{
  "method": "depth.query",
  "params": [
    "BCCBTC",           //1.market: market name
    10,                 //2.limit: Count limit, Integer
    "0"                 //3.interval: Merge, String e.g. "0" for no interval
  ],
  "id": 1517465832
}
```

Response:

```
{
  "error": null,
  "result": {
    "asks": [           //Depth of Sell
      [
        "12.94",       //Sell out price
        "0.1524"       //Sell out count
      ]
    ]
  },
  "id": 1517465832
}
```

2.3.2 Subscribe market depth**Example request:**

```
{
  "method": "depth.subscribe",
  "params": [
    "BCCBTC",           //1.market: market name
    10,                 //2.limit: Count limit, Integer
    "0"                 //3.interval: String, e.g. "0" for no interval
  ],
  ↪markets
  "id": 1517465832
}
```

Notify:

```
{
  "method": "depth.update",
  "params": [
    true,               //Boolean, true: for complete result, false: for ↪
    ↪update based on latest retrun result
    {                   //Update info
      "bids": [         //Depth of Buy
        [
          "0.21082",    //Buy in price
          "0.0588"      //Buy in count
        ]
      ],
      "asks": [         //Depth of Sell
        [
          "0.25082",    //Sell out price
          "0.988"       //Sell in count
        ]
      ]
    }
  ],
  "BCCBTC"
  "id": null
}
```

2.3.3 Unsubscribe depth

Example request:

```
{
  "method": "depth.unsubscribe",
  "params": [],
  "id": 1517466483
}
```

Response:

```
{
  "error": null,
  "result": {
    "status": "success"
  },
  "id": 1517466483
}
```

2.4 Price API

2.4.1 Acquire latest price

Example request:

```
{
  "method": "price.query",
  "params": [
    "BCCBTC" //1. market: String, market name
  ],
  "id": 1517466483
}
```

Response:

```
{
  "error": null,
  "id": 1517466483,
  "result": "0.14900000"
}
```

2.4.2 Latest price subscription

Example request:

```
{
  "method": "price.subscribe",
  "params": [
    "BCCBTC" //1. market list
  ],
  "id": 1517466483
}
```


notify:

```
{
  "method": "price.update",
  "params": [
    "BCCBTC",
    "0.149"
  ],
  "id": null
}
```

2.4.3 Latest price unsubscribe

Example request:

```
{
  "method": "price.unsubscribe",
  "params": [],
  "id": 1517466483
}
```

Response:

```
{
  "error": null,
  "result": {
    "status": "success"
  },
  "id": 1517466483
}
```

2.5 Deal API

2.5.1 Acquire latest executed list

Example request:

```
{
  "method": "deals.query",
  "params": [
    "BCCBTC",           //1.market: market name
    10,                 //2.limit: amount limit
    7561                //3.last_id: largest ID of last returned result
  ],
  "id": 1223
}
```

Response:

```
{
  "error": null,
  "id": 1223,
  "result": []
}
```

2.5.2 Latest order list subscription

Example request:

```
{
  "method": "deals.subscribe",
  "params": [
    #1. params: market list
    "BCCBTC",
    "BCDBTC"
  ],
  "id": 1223
}
```

Notify:

```
{
  "method": "deals.update",
  "params": [
    "BCDBTC",          //1.market name
    [                  //2.order list
      {
        "price": "0.000801",
        "id": 6834,
        "time": 1517370019.46558,
        "amount": "10.5",
        "type": "buy"
      },
      {
        "price": "0.000801",
        "id": 6833,
        "time": 1517369971.09925,
        "amount": "4.5",
        "type": "sell"
      }
    ]
  ],
  "id": null
}
```

2.5.3 Cancel subscription

Example request:

```
{
  "method": "deals.unsubscribe",
  "params": [],
  "id": 1517466483
}
```

Response:

```
{
  "error": null,
  "result": {
    "status": "success"
  },
}
```

```
"id": 1517466483  
}
```