# Ccwt\_web: ccwt\_web接口的数据没有进行转换，直接从MongoDB读取；

## Host: http://52.194.88.72:8002

### Kline: /api/kline

参数：

exchange: 交易所 str

symbol: 交易对，可以多个，以逗号分隔 str

time\_frame: 时间范围（1m、1d，1w, 1M），默认1m str

limit: 条数,默认100条 str

format: 返回数据格式, 默认API格式， （json: json格式， api: api格式） str

<http://52.194.88.72:8002/api/kline?exchange=bitmex&symbol=TRXZ18&time_frame=1m&limit=10>

[

{

"bitmex\_TRXZ18\_kline\_1m": [

{

"\_id": "1540369140.0",

"volume": 1910.0,

"vwap": 3.7e-06,

"turnover": 706700,

"sys\_time": "2018-10-24 08:19:15",

"open": 3.7e-06,

"date": 20181024,

"home\_notional": null,

"low": 3.7e-06,

"last\_size": 10,

"high": 3.7e-06,

"close": 3.7e-06,

"exchange\_time": "2018-10-24 08:19:00.000000",

"count": 2,

"foreign\_notional": null,

"time\_stamp": "2018-10-24T08:19:00"

},

{

"\_id": "1540369320.0",

"date": 20181024,

"foreign\_notional": null,

"volume": 40000.0,

"low": 3.71e-06,

"last\_size": 29900,

"time\_stamp": "2018-10-24T08:22:00",

"turnover": 14840000,

"sys\_time": "2018-10-24 08:22:15",

"exchange\_time": "2018-10-24 08:22:00.000000",

"open": 3.7e-06,

"high": 3.71e-06,

"vwap": 3.71e-06,

"home\_notional": null,

"count": 3,

"close": 3.71e-06

},

]

}

]

### Ticker: /api/ticker

参数：

exchange: 交易所 str

symbol: 交易对，可以多个，以逗号分隔 str

limit: 条数,默认100条 str

format: 返回数据格式, 默认API格式， （json: json格式， api: api格式） str

http://52.194.88.72:8002/api/ticker?exchange=bitmex&symbol=TRXZ18&limit=10

[

{

"bitmex\_TRXZ18\_ticker": [

{

"\_id": "2018-10-23T08:33:01.866Z",

"symbol": "TRXZ18",

"rootSymbol": "TRX",

"state": "Open",

"typ": "FFCCSX",

"listing": "2018-09-21T02:00:00.000Z",

"front": "2018-11-30T12:00:00.000Z",

"expiry": "2018-12-28T12:00:00.000Z",

"settle": "2018-12-28T12:00:00.000Z",

"relistInterval": null,

"inverseLeg": "",

"sellLeg": "",

"buyLeg": "",

"optionStrikePcnt": null,

"optionStrikeRound": null,

"optionStrikePrice": null,

"optionMultiplier": null,

"positionCurrency": "TRX",

"underlying": "TRX",

"quoteCurrency": "XBT",

"underlyingSymbol": "TRXXBT=",

"reference": "BMEX",

"referenceSymbol": ".TRXXBT30M",

"calcInterval": null,

"publishInterval": null,

"publishTime": null,

"maxOrderQty": 10000000,

"maxPrice": 100,

"lotSize": 1,

"tickSize": 1e-08,

"multiplier": 100000000,

"settlCurrency": "XBt",

"underlyingToPositionMultiplier": 1,

"underlyingToSettleMultiplier": null,

"quoteToSettleMultiplier": 100000000,

"isQuanto": false,

"isInverse": false,

"initMargin": 0.05,

"maintMargin": 0.025,

"riskLimit": 5000000000,

"riskStep": 5000000000,

"limit": null,

"capped": false,

"taxed": true,

"deleverage": true,

"makerFee": -0.0005,

"takerFee": 0.0025,

"settlementFee": 0,

"insuranceFee": 0,

"fundingBaseSymbol": "",

"fundingQuoteSymbol": "",

"fundingPremiumSymbol": "",

"fundingTimestamp": null,

"fundingInterval": null,

"fundingRate": null,

"indicativeFundingRate": null,

"rebalanceTimestamp": null,

"rebalanceInterval": null,

"openingTimestamp": "2018-10-23T08:00:00.000Z",

"closingTimestamp": "2018-10-23T09:00:00.000Z",

"sessionInterval": "2000-01-01T01:00:00.000Z",

"limitDownPrice": null,

"limitUpPrice": null,

"bankruptLimitDownPrice": null,

"bankruptLimitUpPrice": null,

"prevTotalVolume": 15128735716,

"totalVolume": 15129304596,

"volume24h": 288268049,

"prevTotalTurnover": 5709724915753,

"totalTurnover": 5709934517131,

"turnover": 209601378,

"turnover24h": 107140555974,

"homeNotional24h": 288268049,

"foreignNotional24h": 1071.4055597400004,

"prevPrice24h": 3.77e-06,

"vwap": 3.72e-06,

"lastPriceProtected": 3.69e-06,

"lastTickDirection": "ZeroPlusTick",

"lastChangePcnt": -0.0212,

"midPrice": 3.685e-06,

"impactBidPrice": 3.68e-06,

"impactMidPrice": 3.685e-06,

"impactAskPrice": 3.69e-06,

"hasLiquidity": true,

"openInterest": 316012128,

"openValue": 116292463104,

"fairMethod": "ImpactMidPrice",

"fairBasisRate": 0.1,

"fairBasis": 7e-08,

"fairPrice": 3.68e-06,

"markMethod": "FairPrice",

"markPrice": 3.68e-06,

"indicativeTaxRate": 0,

"indicativeSettlePrice": 3.61e-06,

"optionUnderlyingPrice": null,

"settledPrice": null,

"timestamp": "2018-10-23T08:33:01.866Z",

"high": 3.79e-06,

"low": 3.64e-06,

"close": 3.69e-06,

"preclose": 3.77e-06,

"bid": 3.68e-06,

"ask": 3.69e-06,

"base\_volume": 568880,

"sys\_time": "2018-10-23 16:33:04",

"date": "20181023"

},]

}

]

### Depth: /api/depth

参数：

exchange: 交易所 str

symbol: 交易对，可以多个，以逗号分隔 str

limit: 条数,默认100条 str

format: 返回数据格式, 默认API格式， （json: json格式， api: api格式） str

<http://52.194.88.72:8002/api/depth?exchange=binance&symbol=ADABTC&limit=10>

[

{

"binance\_ADABTC\_depth": [

{

"\_id": "95474933",

"last\_ID": 95474933,

"bids": [

[

1.004e-05,

1.004e-05

],

[

1.003e-05,

1.003e-05

],

[

1.002e-05,

1.002e-05

],

[

1.001e-05,

1.001e-05

],

[

1e-05,

1e-05

],

[

9.99e-06,

9.99e-06

],

[

9.98e-06,

9.98e-06

],

[

9.97e-06,

9.97e-06

],

[

9.96e-06,

9.96e-06

],

[

9.95e-06,

9.95e-06

],

[

9.94e-06,

9.94e-06

],

[

9.93e-06,

9.93e-06

],

[

9.92e-06,

9.92e-06

],

[

9.91e-06,

9.91e-06

],

[

9.9e-06,

9.9e-06

],

[

9.89e-06,

9.89e-06

],

[

9.88e-06,

9.88e-06

],

[

9.87e-06,

9.87e-06

],

[

9.86e-06,

9.86e-06

],

[

9.85e-06,

9.85e-06

]

],

"asks": [

[

1.006e-05,

1.006e-05

],

[

1.007e-05,

1.007e-05

],

[

1.008e-05,

1.008e-05

],

[

1.009e-05,

1.009e-05

],

[

1.01e-05,

1.01e-05

],

[

1.011e-05,

1.011e-05

],

[

1.012e-05,

1.012e-05

],

[

1.013e-05,

1.013e-05

],

[

1.014e-05,

1.014e-05

],

[

1.015e-05,

1.015e-05

],

[

1.016e-05,

1.016e-05

],

[

1.017e-05,

1.017e-05

],

[

1.018e-05,

1.018e-05

],

[

1.019e-05,

1.019e-05

],

[

1.02e-05,

1.02e-05

],

[

1.021e-05,

1.021e-05

],

[

1.022e-05,

1.022e-05

],

[

1.023e-05,

1.023e-05

],

[

1.024e-05,

1.024e-05

],

[

1.025e-05,

1.025e-05

]

],

"date": "20180918",

"sys\_time": "2018-09-18 09:09:45"

},]

}

]

### Order: /api/order

参数：

exchange: 交易所 str

symbol: 交易对，可以多个，以逗号分隔 str

limit: 条数,默认100条 str

format: 返回数据格式, 默认API格式， （json: json格式， api: api格式） str

<http://52.194.88.72:8002/api/order?exchange=binance&symbol=ADABTC&limit=10>

[

{

"binance\_ADABTC\_order": [

{

"\_id": "1537232981738",

"time\_stamp": 1537232981738,

"first\_ID": "95474920",

"last\_ID": "95474920",

"bids": [],

"asks": [

[

1.055e-05,

1.055e-05

]

],

"exchange\_time": "2018-09-18 01:09:41.000000",

"date": "20180918",

"sys\_time": "2018-09-18 09:09:41"

},]

}

]

### Trade: /api/trade

参数：

exchange: 交易所 str

symbol: 交易对，可以多个，以逗号分隔 str

limit: 条数,默认100条 str

format: 返回数据格式, 默认API格式， （json: json格式， api: api格式） str

[

{

"binance\_ADABTC\_trade": [

{

"\_id": "14793414",

"time\_stamp": 1537232982570,

"order": "null",

"type": "null",

"side": "buy",

"taker\_or\_maker": "null",

"price": 1.006e-05,

"amount": 6.0,

"cost": 6.036e-05,

"fee\_cost": 0.0,

"fee\_currency": "null",

"fee\_rate": 0.0,

"exchange\_time": "2018-09-18 01:09:42.000000",

"date": "20180918",

"sys\_time": "2018-09-18 09:09:42"

},]

}

]

# Ccwt\_client: ccwt\_web客户端:

下载ccwt\_client包：pip install ccwt\_client，下载过的可进行更新 pip install ccwt\_client -U

## Ccwt\_client.core.py

### Cli.tieker(\*\*kwargs):

from ccwt\_client.core import cli  
params = {  
 **'exchange'**: **'bitmex'**, **'symbol'**: **'XBTUSD'**, **'limit'**: **'1'** }  
res = cli.ticker(\*\*params)  
print(res)

[{'bitmex\_XBTUSD\_ticker': [{'\_id': '2018-10-23T08:20:16.103Z', 'symbol': 'XBTUSD', 'rootSymbol': 'XBT', 'state': 'Open', 'typ': 'FFWCSX', 'listing': '2016-05-13T12:00:00.000Z', 'front': '2016-05-13T12:00:00.000Z', 'expiry': None, 'settle': None, 'relistInterval': None, 'inverseLeg': '', 'sellLeg': '', 'buyLeg': '', 'optionStrikePcnt': None, 'optionStrikeRound': None, 'optionStrikePrice': None, 'optionMultiplier': None, 'positionCurrency': 'USD', 'underlying': 'XBT', 'quoteCurrency': 'USD', 'underlyingSymbol': 'XBT=', 'reference': 'BMEX', 'referenceSymbol': '.BXBT', 'calcInterval': None, 'publishInterval': None, 'publishTime': None, 'maxOrderQty': 10000000, 'maxPrice': 1000000, 'lotSize': 1, 'tickSize': 0.5, 'multiplier': -100000000, 'settlCurrency': 'XBt', 'underlyingToPositionMultiplier': None, 'underlyingToSettleMultiplier': -100000000, 'quoteToSettleMultiplier': None, 'isQuanto': False, 'isInverse': True, 'initMargin': 0.01, 'maintMargin': 0.005, 'riskLimit': 20000000000, 'riskStep': 10000000000, 'limit': None, 'capped': False, 'taxed': True, 'deleverage': True, 'makerFee': -0.00025, 'takerFee': 0.00075, 'settlementFee': 0, 'insuranceFee': 0, 'fundingBaseSymbol': '.XBTBON8H', 'fundingQuoteSymbol': '.USDBON8H', 'fundingPremiumSymbol': '.XBTUSDPI8H', 'fundingTimestamp': '2018-10-23T12:00:00.000Z', 'fundingInterval': '2000-01-01T08:00:00.000Z', 'fundingRate': 0.0001, 'indicativeFundingRate': 0.0001, 'rebalanceTimestamp': None, 'rebalanceInterval': None, 'openingTimestamp': '2018-10-23T08:00:00.000Z', 'closingTimestamp': '2018-10-23T09:00:00.000Z', 'sessionInterval': '2000-01-01T01:00:00.000Z', 'limitDownPrice': None, 'limitUpPrice': None, 'bankruptLimitDownPrice': None, 'bankruptLimitUpPrice': None, 'prevTotalVolume': 866877370056, 'totalVolume': 866879525586, 'volume24h': 584437921, 'prevTotalTurnover': 11862857877788048, 'totalTurnover': 11862891578522378, 'turnover': 33700734329, 'turnover24h': 9131853536643, 'homeNotional24h': 91318.53536643005, 'foreignNotional24h': 584437921, 'prevPrice24h': 6403, 'vwap': 6400, 'lastPriceProtected': 6396.3157, 'lastTickDirection': 'ZeroPlusTick', 'lastChangePcnt': -0.001, 'midPrice': 6396.25, 'impactBidPrice': 6395.9066, 'impactMidPrice': 6396, 'impactAskPrice': 6396.3157, 'hasLiquidity': True, 'openInterest': 763308087, 'openValue': 11930505399810, 'fairMethod': 'FundingRate', 'fairBasisRate': 0.1095, 'fairBasis': 0.29, 'fairPrice': 6397.95, 'markMethod': 'FairPrice', 'markPrice': 6397.95, 'indicativeTaxRate': 0, 'indicativeSettlePrice': 6397.66, 'optionUnderlyingPrice': None, 'settledPrice': None, 'timestamp': '2018-10-23T08:20:16.103Z', 'high': 6423, 'low': 6369, 'close': 6396.5, 'preclose': 6399.85, 'bid': 6396, 'ask': 6396.5, 'base\_volume': 2155330, 'sys\_time': '2018-10-23 16:20:19', 'date': '20181023', 'volume': 2155530}], 'cached': False}]

### Cli.kline(\*\*kwargs):

from ccwt\_client.core import cli  
params = {  
 **'exchange'**: **'bitmex'**, **'symbol'**: **'XBTUSD'**, **'limit'**: **'1'** }  
res = cli.kline(\*\*params)  
print(res)

[{'bitmex\_XBTUSD\_kline\_1m': [{'\_id': '1540369140.0', 'close': 6437.5, 'open': 6438.0, 'count': 13, 'turnover': 776697187, 'sys\_time': '2018-10-24 08:19:15', 'time\_stamp': '2018-10-24T08:19:00', 'low': 6437.5, 'date': 20181024, 'foreign\_notional': None, 'high': 6438.0, 'last\_size': 556, 'exchange\_time': '2018-10-24 08:19:00.000000', 'volume': 50003.0, 'vwap': 6437.9064, 'home\_notional': None}], 'cached': False}]

### Cli.depth(\*\*kwargs):

from ccwt\_client.core import cli  
params = {  
 **'exchange'**: **'binance'**, **'symbol'**: **'ZILBTC'**, **'limit'**: **'1'** }  
res = cli.depth(\*\*params)  
print(res)

[{'binance\_ZILBTC\_depth': [{'\_id': '53744627', 'last\_ID': 53744627, 'bids': [[4.97e-06, 4.97e-06], [4.95e-06, 4.95e-06], [4.94e-06, 4.94e-06], [4.93e-06, 4.93e-06], [4.92e-06, 4.92e-06], [4.91e-06, 4.91e-06], [4.9e-06, 4.9e-06], [4.89e-06, 4.89e-06], [4.88e-06, 4.88e-06], [4.87e-06, 4.87e-06], [4.86e-06, 4.86e-06], [4.85e-06, 4.85e-06], [4.84e-06, 4.84e-06], [4.83e-06, 4.83e-06], [4.82e-06, 4.82e-06], [4.81e-06, 4.81e-06], [4.8e-06, 4.8e-06], [4.79e-06, 4.79e-06], [4.78e-06, 4.78e-06], [4.77e-06, 4.77e-06]], 'asks': [[4.98e-06, 4.98e-06], [4.99e-06, 4.99e-06], [5e-06, 5e-06], [5.01e-06, 5.01e-06], [5.02e-06, 5.02e-06], [5.03e-06, 5.03e-06], [5.04e-06, 5.04e-06], [5.05e-06, 5.05e-06], [5.06e-06, 5.06e-06], [5.07e-06, 5.07e-06], [5.08e-06, 5.08e-06], [5.09e-06, 5.09e-06], [5.1e-06, 5.1e-06], [5.11e-06, 5.11e-06], [5.12e-06, 5.12e-06], [5.13e-06, 5.13e-06], [5.14e-06, 5.14e-06], [5.15e-06, 5.15e-06], [5.16e-06, 5.16e-06], [5.17e-06, 5.17e-06]], 'date': '20180918', 'sys\_time': '2018-09-18 09:09:44'}], 'cached': False}]

### Cli.order(\*\*kwargs):

from ccwt\_client.core import cli  
params = {  
 **'exchange'**: **'binance'**, **'symbol'**: **'ZILBTC'**, **'limit'**: **'1'** }  
res = cli.order(\*\*params)  
print(res)

[{'binance\_ZILBTC\_order': [{'\_id': '1537232981739', 'time\_stamp': 1537232981739, 'first\_ID': '53744625', 'last\_ID': '53744625', 'bids': [[4.91e-06, 4.91e-06]], 'asks': [], 'exchange\_time': '2018-09-18 01:09:41.000000', 'date': '20180918', 'sys\_time': '2018-09-18 09:09:42'}], 'cached': False}]

### Cli.trade(\*\*kwargs):

from ccwt\_client.core import cli  
params = {  
 **'exchange'**: **'binance'**, **'symbol'**: **'ZILBTC'**, **'limit'**: **'1'** }  
res = cli.trade(\*\*params)  
print(res)

[{'binance\_ZILBTC\_trade': [{'\_id': '5014013', 'time\_stamp': 1537232986458, 'order': 'null', 'type': 'null', 'side': 'buy', 'taker\_or\_maker': 'null', 'price': 4.98e-06, 'amount': 4254.0, 'cost': 0.02118492, 'fee\_cost': 0.0, 'fee\_currency': 'null', 'fee\_rate': 0.0, 'exchange\_time': '2018-09-18 01:09:46.000000', 'date': '20180918', 'sys\_time': '2018-09-18 09:09:46'}], 'cached': False}]

## Ccwt\_client.core\_feed.py:

数据格式支持pyalgotrade框架

feed = Feed(bar.Frequency.SECOND)

“””

SECOND = 1  
 MINUTE = 60  
 HOUR = 60\*60  
 DAY = 24\*60\*60  
 WEEK = 24\*60\*60\*7  
 MONTH = 24\*60\*60\*31

“””

feed.loadBars(**"bitmex\_XBTUSD"**)

数据格式：

\_\_slots\_\_ = (  
 **'\_\_dateTime'**,  
 **'\_\_open'**,  
 **'\_\_close'**,  
 **'\_\_high'**,  
 **'\_\_low'**,  
 **'\_\_volume'**,  
 **'\_\_adjClose'**,  
 **'\_\_frequency'**,  
 **'\_\_useAdjustedValue'**,  
 **'\_\_extra'**,  
)

[[datetime.datetime(2018, 10, 23, 8, 20, 58, 136000), 6399.85, 6423, 6369, 6396, 2155330, None, 1], [datetime.datetime(2018, 10, 23, 8, 20, 16, 103000), 6399.85, 6423, 6369, 6396.5, 2155330, None, 1], [datetime.datetime(2018, 10, 23, 7, 19, 9, 591000), 6399.85, 6423, 6369, 6401.5, 6151444, None, 1],]

# Ccwt\_client V0.6 更新：

## ccwt\_feed.py 中Feed类下的loadBars方法修改：

**loadBars方法中，增加test\_back回测标识；test\_back为必输字段，为True时表示获取回测数据，默认获取1000条数据；为False时，start\_date或end\_date不能同时为空；**

ccwt\_feed.py

class Feed():

def loadBars(self, instrument, **test\_back**, timezone='', start\_date='', end\_date=''):

""" 获取交易所ticker/kline数据

:param instrument:

**:param test\_back: 回测标识，True: 表示回测，默认返回1000条数据， False: 获取时间段区间数据**

:param timezone: 时区

**:param start\_date: 开始日期 与系统时间相比，不与交易所时间比较**

**:param end\_date: 截止日期 与系统时间相比，不与交易所时间比较**

:return:

"""

if not test\_back:

if not start\_date and not end\_date:

raise NotImplementedError('test\_back is False, start\_date and end\_date not is empty!')

# log.info("instrument: {}.".format(instrument))

bars = self.db.getBars(instrument, self.getFrequency(), test\_back, timezone, start\_date, end\_date)

self.addBarsFromSequence(instrument, bars)

## ccwt\_client程序结构进行修改：

**strategy模块移到ccwt\_client模块下；**

