Margin Trading - WebSocket Private-data Update Messages

> ℹ️ More information on WebSocket can be found here -
>
> V1: https://api.exchange.bullish.com/docs/api/rest/trading-api/v1/#overview--websockets
>
> V2: https://api.exchange.bullish.com/docs/api/rest/trading-api/v2/#overview--websockets

Goal

The goal of this document is to describe the **update messages** published to WebSocket's **Private Data topics** specific to **Margin Trading in Bullish** for **Customers who are borrowing**

It also details how the numbers in each topic can be reconciled against one another.

Scope

Private Data Updates Only

In general, WebSocket Messages can be divided into the below categories

	Snapshots	Updates
Private Data (Authenticated)	Snapshots for Private Data	Updates for Private Data
Market Data (Both Authenticated and Unauthenticated)	Snapshots for Market Data	Updates for Market Data

The scope of this document is confined to the **updates for private-data** for **customers who are borrowing.**

Private Data Topics

⚠ The `spotAccounts` topic is deprecated and is superseded by the `assetAccounts` topic. The latter provides more comprehensive details regarding the assets in the customer's trading accounts.

There are currently 5 topics available under Private Data:

- orders
- trades
- (Deprecated) spotAccounts
- assetAccounts
- tradingAccounts

Details of the fields of each topic are available in https://api.exchange.bullish.com/docs/api/rest/trading-api/v1/#overview--receive-a-message-from-the-web-socket

Fields Comparison - `spotAccounts` Topic vs `assetAccounts` Topic

With `spotAccounts` topic deprecated in favour of the `assetAccounts` topic, this section shows

- the fields in the assetAccount topic
- a diagram which maps each field in spotAccount topic to assetAccount topic
- a table which lists the formula for each field in spotAccount topic in terms of the fields in assetAccount

assetAccounts Topic

The assetAccounts Topic is a new one and contains the below fields.

Name	Description
tradingAccountId	id of the trading account
assetId	asset ID
assetSymbol	asset symbol
availableQuantity	the assets that are available to use on the account including borrowed assets, see asset value format
borrowedQuantity	the assets on the account that are borrowed, including interest charged but excluding defaulted amount, see asset value format
lockedQuantity	the assets on the account that are locked in open orders, see asset value format
loanedQuantity	the assets on the account that are in open loan instructions, including interest earned but excluding defaulted amount, see asset value format
updatedAtDatetime	denotes the time the asset account was updated by the exchange, ISO 8601 with millisecond as string

updatedAtTime stamp	denotes the time the asset account was updated by the exchange

Field Mapping - Diagram

spotAccounts

assetAccounts

tradingAccountId

accountId

assetId

symbol

assetSymbol

availableQuantity

total

borrowedQuantity

free

lockedQuantity

loanedQuantity

used

updatedAtDatetime

updatedAtTimestamp
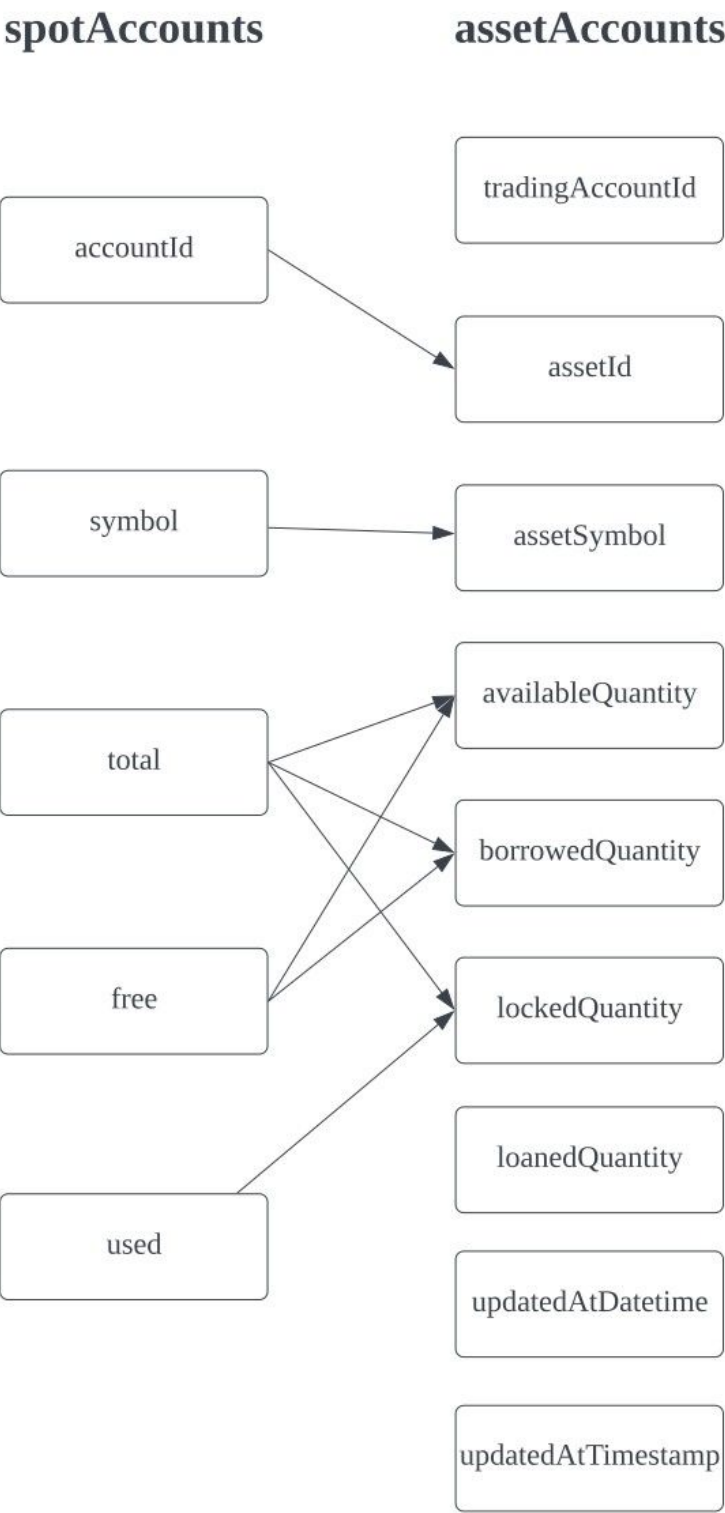
Overview - the `assetAccounts` topic has more fields and is more comprehensive.

Field Mapping - Formula

A numeric example showing the conversion of the fields is available in the section `2.b.iii.When said Aggressive Limit Order is executed` under `Messages`

Fields in spotAccounts topic	Corresponding fields in assetAccounts topic
accountId	assetId
symbol	assetSymbol
total	max(0, availableQuantity - borrowedQuantity) + lockedQuantity
free	max(0, availableQuantity - borrowedQuantity)
used	lockedQuantity

Each field in the `spotAccounts` topic can be derived from the `assetAccounts` topic.

Messages

1. When Trading Account has Max Initial Leverage Greater Than One

> ℹ️ To get the Max Initial Leverage of a trading account, the `GET /accounts/trading-accounts` REST API endpoint provides a field called `maxInitialLeverage`.
> (Link: https://api.exchange.bullish.com/docs/api/rest/trading-api/v2/#get-/accounts/trading-accounts)

Topics	What happens?	Frequency	Example Message
trades	N/A		
orders	N/A		

tradingAccounts	An update is published once every 30 seconds on the trading account	Recurring.
		
		Once every 30 seconds
			``json
			{
			    "type": "update",
			    "dataType": "V1TATradingAccount",
			    "data": {
			        "tradingAccountId": "111904161762538",
			        "referenceAssetSymbol": "USD",
			        "totalBorrowedQuantity": "15107.3159",
			        "totalCollateralQuantity": "21099.2971",
			        "updatedAtDatetime": "2023-07-25T07:10:11.16
			        "updatedAtTimestamp": "1690269011169"
			    }
			``

assetAccounts	N/A		
spotAccounts	N/A		

2. When Trading Account is in Liquidation

2.a. When Margin Call Occurs

- Leverage reaches 5x

Topics	What happens?	Frequency	Example Message
trades	N/A		
orders	N/A		
tradingAccounts	N/A		
assetAccounts	N/A		
spotAccounts	N/A		

2.b. When Partial Liquidation Occurs

- Leverage reaches 6x

2.b.i When Trading Account has Open Limit Orders

Topics	What happens?	Frequency	Example Message
trades	N/A		
orders	The open limit order gets cancelled	Once	(see code below)

```
1  {
2      "type": "update",
3      "dataType": "V1TAOrder",
4      "data": {
5          "status": "CANCELLED",
6          "timeInForce": "GTC",
7          "borrowedQuantity": "0.00050100",
8          "baseFee": "0.00000000",
9          "price": "11650.0000",
10         "createdAtTimestamp": "1690267713122",
11         "quoteFee": "0.0000",
12         "statusReason": "Unsolicited cancel",
13         "stopPrice": null,
14         "quantityFilled": "0.00000000",
15         "type": "LMT",
16         "handle": null,
17         "statusReasonCode": 3020,
18         "orderId": "603840999349288961",
19         "quantity": "0.50000000",
20         "margin": true,
21         "side": "SELL",
22         "createdAtDatetime": "2023-07-25T06:48:33.122Z",
23         "isLiquidation": false,
24         "symbol": "BTCUSDC",
```

			25	`"averageFillPrice": null`
			26	`}`
tradingAccounts	N/A			
assetAccounts	N/A			
spotAccounts	N/A			

2.b.ii When an Aggressive Limit Order is created to Partially Repay the Debt

Topics	What happens?	Frequency	Example Message
trades	N/A		
| orders | • With the view to repay the debt partially, the liquidation engine sends an order to partially liquidate the trading account. | Once | ```
1  {   "type": "update",
2      "dataType": "V1TAOrder",
3      "data": {
4          "status": "OPEN",
5          "timeInForce": "GTC",
6          "borrowedQuantity": null,
7          "baseFee": "0.00000000",
8          "price": "17651.4000",
9          "createdAtTimestamp": "1690259245994",
10         "quoteFee": "0.0000",
11         "statusReason": "Open",
12         "stopPrice": null,
13         "quantityFilled": "0.00000000",
14         "type": "LMT",
15         "handle": null,
16         "statusReasonCode": 6001,
17         "orderId": "680",
18         "quantity": "0.16104577",
19         "margin": false,
20         "side": "BUY",
21         "createdAtDatetime": "2023-07-25T04:27:25.994Z",
22         "isLiquidation": true,
23         "symbol": "BTCUSDC",
24         "averageFillPrice": null
25     }
``` |
tradingAccounts	N/A		
assetAccounts	N/A		
spotAccounts	N/A		

2.b.iii When said Aggressive Limit Order is executed

Topics	What happens?	Frequency	Example Message

| trades | The executed order becomes a trade | Once | ```json
{
    "type": "update",
    "dataType": "V1TATrade",
    "data": {
        "tradeId": "100000000000000009",
        "handle": null,
        "baseFee": "0.00000000",
        "isTaker": true,
        "price": "11600.7822",
        "orderId": "680",
        "createdAtTimestamp": "1690259245994",
        "quoteFee": "1.8683",
        "quantity": "0.16104577",
        "side": "BUY",
        "createdAtDatetime": "2023-07-25T04:27:25.994Z",
        "symbol": "BTCUSDC"
    }
}
``` |
|---|---|---|---|
| orders | The aggressive limit order sent by the liquidation engine gets executed | Once | ```json
{
    "type": "update",
    "dataType": "V1TAOrder",
    "data": {
        "status": "CLOSED",
        "timeInForce": "GTC",
        "borrowedQuantity": null,
        "baseFee": "0.00000000",
        "price": "17651.4000",
        "createdAtTimestamp": "1690259245994",
        "quoteFee": "0.0000",
        "statusReason": "Executed",
        "stopPrice": null,
        "quantityFilled": "0.16104577",
        "type": "LMT",
        "handle": null,
        "statusReasonCode": 6002,
        "orderId": "680",
        "quantity": "0.16104577",
        "margin": false,
        "side": "BUY",
        "createdAtDatetime": "2023-07-25T04:27:25.994Z",
        "isLiquidation": true,
        "symbol": "BTCUSDC",
        "averageFillPrice": "11600.7822"
    }

}
``` |
| tradingAccounts | The executed trade changes the totalBorrowedQuantity and totalCollateralQuantity in the trading Account | Once | ```json
{
    "type": "update",
    "dataType": "V1TATradingAccount",
    "data": {
        "tradingAccountId": "111904161762538",
        "referenceAssetSymbol": "USD",
        "totalBorrowedQuantity": "28456.2608",
        "totalCollateralQuantity": "32972.1419",
        "updatedAtDatetime": "2023-07-25T04:27:25.990Z",
        "updatedAtTimestamp": "1690259245990"
``` |

```
11        }
12  }
```

assetAc counts	Balance of BTC (borrowed asset) and USDC (collateral asset) change	Once	

1. <u>USDC balance drops</u> (note how the sum `availableQuantity + lockedQuantity` drops.).
   - USDC Balance just **before** the liquidation order gets executed
     - i. Note:
       1. The reason why availableQuantity increases is that **not all** of the USDC locked in the liquidation order gets sold. However, if we check the sum `availableQuantity + lockedQuantity`, it is clear that USDC balance has dropped.
       2.
     -
       ```
       1  {
       2      "type": "update",
       3      "dataType": "V1TAAssetAccount",
       4      "data": {
       5          "tradingAccountId": "111904161762538",
       6          "assetId": "5",
       7          "assetSymbol": "USDC",
       8          "availableQuantity": "28207.0523",
       9          "borrowedQuantity": "0.0000",
       10         "lockedQuantity": "2859.8396",
       11         "loanedQuantity": "0.0000",
       12         "updatedAtDatetime": "2023-07-25T04:27:25.996Z",
       13         "updatedAtTimestamp": "1690259245996"
       14     }
       15 }
       ```
   - USDC Balance just **after** the liquidation order gets executed
     -
       ```
       1
       2  {
       3      "type": "update",
       4      "dataType": "V1TAAssetAccount",
       5      "data": {
       6          "tradingAccountId": "111904161762538",
       7          "assetId": "5",
       8          "assetSymbol": "USDC",
       9          "availableQuantity": "29187.4254",
       10         "borrowedQuantity": "0.0000",
       11         "lockedQuantity": "0.0000",
       12         "loanedQuantity": "0.0000",
       13         "updatedAtDatetime": "2023-07-25T04:27:25.998Z",
       14         "updatedAtTimestamp": "1690259245998"
       15     }
       16 }
       ```

2. BTC has non-zero **availableQuantity** (to be used to repay **borrowedQuantity**).
   - a. Note:
     - i. `availableQuantity` **Delta =** 0.16104577 (now) - 0 (previously) = `quantity` **in trade topic** = 0.1610457.
     - ii. When the hourly auto-repayment comes, the **availableQuantity** will be used to reduce the **borrowedQuantity**. However, this will not happen right after the liquidation order is executed. That is why we see a non-zero **availableQuantity** here. which could potentially be used to reduce **borrowedQuantity.**

```
1   {
2       "type": "update",
3       "dataType": "V1TAAssetAccount",
4       "data": {
5           "tradingAccountId": "111904161762538",
6           "assetId": "1",
7           "assetSymbol": "BTC",
8           "availableQuantity": "0.16104577",
9           "borrowedQuantity": "1.62824806",
10          "lockedQuantity": "0.00000000",
11          "loanedQuantity": "0.00000000",
12          "updatedAtDatetime": "2023-07-25T04:27:25.998Z",
13          "updatedAtTimestamp": "1690259245998"
14      }
15  }
```

spotAccounts	• Balance of BTC (borrowed asset) and USDC (collateral asset) change	Once	• USDC balance drops

• USDC balance drops
  ○ USDC Balance just before liquidation order gets executed
    i. **used = lockedQuantity = 2859.8396**
    ii. **free = max(0, availableQuantity - borrowedQuantity) = max(0, 28207.0523 - 0) = 28207.0523**
    iii. **total = max(0, availableQuantity - borrowedQuantity) + locked = 28207.0523 + 2859.8396 = 31066.8919**

  ○
```
1   {
2       "type": "update",
3       "dataType": "V1TASpotAccount",
4       "data": {
5           "type": "spot",
6           "accountId": "5",
7           "symbol": "USDC",
8           "total": "31066.8919",
9           "free": "28207.0523",
10          "used": "2859.8396"
11      }
12  }
```

  ○ USDC Balance just after liquidation order gets executed
    ▪ **used = lockedQuantity = 0**
    ▪ **free = max(0, availableQuantity - borrowedQuantity) = max(0, 29187.4254 - 0) = 29187.4254**
    ▪ **total = max(0, availableQuantity - borrowedQuantity) + locked = 29187.4254 + 0 = 29187.4254**

  ○
```
1   {
2       "type": "update",
3       "dataType": "V1TASpotAccount",
4       "data": {
5           "type": "spot",
6           "accountId": "5",
7           "symbol": "USDC",
8           "total": "29187.4254",
9           "free": "29187.4254",
```

```
10          "used": "0.0000"
11      }
12  }
```

2. BTC remains unchanged (Note: We can confirm the values of the fields are correct)

    a. **used** = **lockedQuantity** = 0

    b. **free = max(0, availableQuantity - borrowedQuantity) = max(0, 0.16104577 -**
       1.62824806) = 0

    c. **total = max(0, availableQuantity - borrowedQuantity) + lockedQuantity = max(0,**
       0.16104577 - 1.62824806) -0 = 0

```
1  {
2      "type": "update",
3      "dataType": "V1TASpotAccount",
4      "data": {
5          "type": "spot",
6          "accountId": "1",
7          "symbol": "BTC",
8          "total": "0.00000000",
9          "free": "0.00000000",
10          "used": "0.00000000"
11      }
12  }
```

2.c When Full Liquidation occurs

- Leverage reaches 12x

2.c.i When Trading Account has Insufficient Fund

Topics	What happens?	Frequency	Example Message
trades	N/A		
orders	Because of insufficient fund, the aggressive limit order sent by the liquidation engine cannot be executed.	Recurring. Error Messages stop when sufficient fund is deposited into the trading account, upon which the order gets executed.	```1 {
2   "type": "error",
3   "dataType": "V1TAErrorResponse",
4   "data": {
5   "handle": null,
6   "requestId": "1111",
7   "orderId": "85046",
8   "symbol": "BTCUSDC",
9   "message": "Insufficient balance",
10   "errorCode": 3005,
11   "errorCodeName": "INSUFFICIENT_BALANCE"
12   }
13 }
14
15 {
16   "type": "update",
17   "dataType": "V1TAOrder",
18   "data": {
19   "status": "REJECTED",
20   "timeInForce": "GTC",```

```
21          "borrowedQuantity": null,
22          "baseFee": "0.00000000",
23          "price": "238900.1000",
24          "createdAtTimestamp": "1690351352354",
25          "quoteFee": "-0.0001",
26          "statusReason": "Insufficient balance",
27          "stopPrice": null,
28          "quantityFilled": "0.00000000",
29          "type": "LMT",
30          "handle": null,
31          "statusReasonCode": 3005,
32          "orderId": "85046",
33          "quantity": "0.05317730",
34          "margin": true,
35          "side": "BUY",
36          "createdAtDatetime": "2023-07-26T06:02:32.354Z",
37          "isLiquidation": false,
38          "symbol": "BTCUSDC",
39          "averageFillPrice": null
40      }
41  }
```

trading Accounts	N/A		

assetAccounts	Regular updates of the insufficient collateral assets used in the aggressive limit order sent by the liquidation engine.	Recurring.
		
		Error Messages stop when sufficient fund is deposited into the trading account, upon which the order gets executed.
			• Note that only the collateral asset (USDC) is reported.
			
			``
			1 {
			2 "type": "update",
			3 "dataType": "V1TAAssetAccount",
			4 "data": {
			5 "tradingAccountId": "111904161762538",
			6 "assetId": "5",
			7 "assetSymbol": "USDC",
			8 "availableQuantity": "12716.8680",
			9 "borrowedQuantity": "0.0000",
			10 "lockedQuantity": "0.0000",
			11 "loanedQuantity": "0.0000",
			12 "updatedAtDatetime": "2023-07-26T06:17:52.460Z",
			13 "updatedAtTimestamp": "1690352272460"
			14 }
			15 }
			``

spotAccounts	Regular updates of the insufficient collateral assets used in the aggressive limit order sent by the liquidation engine.	Recurring.
		
		Error Messages stop when sufficient fund is deposited into the trading account, upon which the order gets executed.
			``
			1 {
			2 "type": "update",
			3 "dataType": "V1TASpotAccount",
			4 "data": {
			5 "type": "spot",
			6 "accountId": "5",
			7 "symbol": "USDC",
			8 "total": "0",
			9 "free": "0",
			10 "used": "0.0000"
			11 }
			12 }
			``

- Messages are similar to Partial Liquidation. The only difference is that the liquidation engine tries to repay all the BTC debt (borrowed asset)

2.d When Account gets Defaulted

> ℹ️ To see if a trading account get defaulted, the `GET /accounts/trading-accounts` REST API endpoint provides a field called `isDefaulted` . When True, it means said trading account is defaulted.
> (Link: https://api.exchange.bullish.com/docs/api/rest/trading-api/v2/#get-/accounts/trading-accounts)

- Leverage reaches leverage 30x

Topics	What happens?	Frequency	Example Message
trades	N/A		
orders	N/A		
tradingAccounts	N/A		
assetAccounts	N/A		
spotAccounts	N/A		