Millistream Pushserver

18 June 2020

Information

The Millistream Pushserver is a WebSocket server for use to stream realtime/delayed data to web applications.
It uses websockets version 13, and only supports wss (WebSocket Secure).

Client – Server communication

Communication between server and client is done in json. The server will validate your messages via a provided token which needs to be included in all messages from the client and if the token is not valid or present you will be disconnected. The server will send a ping about every 10 seconds to check that the client is still “alive” and a pong message is expected but not required. The server will also respond to client pings with a pong.

The server can send you quotes (including name, symbol and isin and some performance values), news, trades and orderbooks.
Quotes can be requested by list or instruments, news, trades and orderbooks only per instrument, and can be unsubscribed in the same way, and your subscriptions can have an requestid if you need that, all updates on instruments where instruments are affected by a requestid will contain this id, if multiple requests affect an instrument all ids will be included comma-separated. The server will respond to all requests with an image of all data requested and then keep sending updates until an unsubscribe or disconnect is received.

Data types

Type	Description
Numeric	Sequence of digits with optional decimal point and sign character. (ASCII characters '-', '0' – '9' and '.').
String	Free format string in UTF-8 encoding
Date	Date represented in Universal Time Coordinated (UTC) in one of the following format: YYYY-MM-DD
Time	Time represented in Universal Time Coordinated (UTC) as HH:MM:SS, HH:MM:SS.mmm or HH:MM:SS.nnnnnnnnn
Tabular String	A string of pipe (|) separated rows of space separated columns. I.e: “1 2 3|4 5 6” defines a set of two rows with three columns each. If a column contains a space, back-slash or pipe character then it's escaped with a back-slash (\). Absent columns are to be decoded as NULL values, I.e here column 2 from the previous example is NULL in the first row: “1 3|4 5 6”
Bitfield	Unsigned integer which forms a binary bit field. For example the value 3 indicates that the flags #1 and #2 is set while the value 4 indicates that only flag #3 is set. To make it easier we always specify the flag value (I.e “4” instead of “flag #3”) in the description.
Array Numeric	A json array with numeric values
uint	Unsigned integer

Client messages

Subscribe

The subscribe message is formatted as follows (with your own provided token):
{ “token”: “e50c2ed0-dc11-4b4c-aded-97143920dfd9”, “request”: [] }
The requestsarray is filled up with your subscriptionparameters, even though it is an array only one request is supported per message. Each request will only support up to 1000 instruments, if the request will result in more than that you will receive an error 413 Payload to large.

The “request” array can have the following properties:

Property		Type	Mandatory
marketplace	The marketplace to subscribe, requires instrumenttype to be set. Can be comma-separated, (insrefs will be provided by Millistream)	uint or string**	*
list	The list to subscribe to, can be comma-separated, (insrefs will be provided by Millistream)	uint or string**	*
insrefs	Instrument(s) to subscribe to, can be comma-separated, (insrefs will be provided by Millistream)	uint or string**	*
instrumenttype	Set which instrumenttypes to request	uint or string**	***

instrumentsubtype	Set which instrumentsubtypes to request	uint or string**	no
mc	Messagetype to request	uint	yes
maxrows	Maxrows to include in the image (orderbooks and trades)	uint	no
id	Your own request id	uint****	yes
unsubscribe	Set to true to unsubscribe request id	boolean	no

* One of marketplace, list or insrefs must be sent in the request
** If comma separated it has to be a string
*** Mandatory if your request is based on marketplace
**** id has to be greater than 0

Messagetypes (“mc”):

Type	**Arrayname**	**Value**
Newsheadlines	newsheadline	1
Quotes	quote	2
Trades	trade	4
Orderbooks	order, orderbookflush	8
Basicdata	basicdata	16
~~Newscontent~~	~~newscontent~~	~~64~~
Performance	performance	1024
Netorderimbalance	noii	16384

Messagetypes can be or:d together so to get basicdata and quotes you will get an mc of 18.

Example request to subscribe to quotes and basicdata for Nasdaq OMX Large Cap Sweden list (insref 35207) with requestid 1:
{“token”:“e50c2ed0-dc11-4b4c-aded-97143920dfd9”,“request”:[{“list”:“35207”,”mc”:”18”,”id”:1}] }

Unsubscribe

Unsubscribe is sent with the requestid you want to unsubscribe and “unsubscribe”:true

Example of unsubscribe of the request above:
{“token”:“e50c2ed0-dc11-4b4c-aded-97143920dfd9”,“request”:[{”id”:1,”unsubscribe”: true }] }

When a client has no subscriptions left after an unsubscribe the server will disconnect the client after a short while if no new requests are sent.

Server messages

The server will not send any messages until it has received a request (ping and pong will still be sent and responded to). When a subscription request has been received it will respond with an image for that request and then start streaming the updates accordingly. Each response will have one array, “instruments”, where each instrument will have an insref and a requestid (if specified in a request) and array(s) with updated data.

Data arrays

quote:

The quote array can have the following properties in it:

Property	**Description**	**Format**
3	Date of the update	Date

4	UTC time of the update	Time
5	Bid price	Numeric
6	Ask price	Numeric
7	Last price	Numeric
8	Dayhighprice	Numeric
9	Daylowprice	Numeric
10	Quantity	Numeric
11	Turnover	Numeric
19	Bid quantity	Numeric
20	Ask quantity	Numeric
21	Symbol	String
22	Name	String
23	Isin	String
37	Number of trades	Numeric
39	Openprice	Numeric
74	Bidyield	Numeric
75	Askyield	Numeric
76	Lastyield	Numeric
77	Openyield	Numeric
78	Dayhighyield	Numeric
79	Daylowyield	Numeric
120	Unchangedpaid	Numeric
121	Pluspaid	Numeric
122	Minuspaid	Numeric
123	Vwap	Numeric
1024	Diff price 1 day	Numeric
1025	Diff price 1 day percent	Numeric
1026	Diff yield 1 day	Numeric
1027	Diff yield 1 day percent	Numeric

The response for the request above will look something like this (with more instruments):

```
{"instruments": [{"requestid": 1,"insref ": 772,
"basicdata": [{"21": "ERIC B","22": "Ericsson B","23": "SE0000108656"}],
"quote": [{"4": "12:48:32","7": "122.20","10": "388094","11": "46732918.45","37": "1469","123":
"120.38074802","1024": "4.300000","1025": "3.647159"}]}]}
```

and an update message will look like this:

```
{"instruments": [{"requestid": 1,"insref": 772,"quote": [{"4": "12:48:32","7": "122.20","10": "388094 ","11":
"46732918.45","37": "1469 ","123": "120.38074802 ","1024": "4.300000 ","1025": "3.647159 "}]}]}
```

newsheadline:

The newsheadline array can have the following properties in it:

Property	Description	Format
0	Language according to ISO 639-1	String

1	Headline	String
3	Date of the update	Date
4	UTC time of the update	Time
21	Symbol of source	String
22	Source Name	String
48	News id	String
86	Newscodingtype ● 0 – News Flash ● 1 – News Article ● 2 – Press Release ● 3 – Company Financial Calendar ● 4 – Report ● 5 – Market Commentary ● 6 – Economic Calendar (Today) ● 7 – Economic Calendar (This Week) ● 8 – Systems Messages	Numeric
167	Newscoding, array of insrefs	Array Numeric
584	Entitlementpackage	Numeric
620	Newscodingtags (Specifies the tags for a news item, single column and each row defines a new tag for the same news item.For supported values please see the "News Format" documentation)	Tabular String
1101	Newscodingmarketplace	Array Numeric
1102	Newscodinglist	Array Numeric

The newsheadline message will look something like this:

{"instruments":[{"requestid":2,"insref":8212,"newsheadline":[{"1":"News headline","3":"2018-04-27", "4":"14:34:00","21":"FWS","48":"6f32bc1e-8c0a-4e61-9d36-144226eefdca"}]}]}

trade:

The trade array can have the following properties in it:

Property	**Description**	**Format**
3	Date of the update	Date
12	Trade price	Numeric
13	Trade quantity	Numeric
14	Trade reference, a reference that uniquely identifies this trade for this instrument on this date.	String
15	Trade code Specifies the trade condition, valid values: ● 1024 – Dark Pool ● 512 – Delayed Dissemination ● 256 – Odd Lot ● 128 – Trade updates Last Price ● 64 – Trade updates Quantity and Turnover ● 32 – Trade updates Day High Price and Day Low Price ● 16 – Trade Break (canceled trade) ● 8 – Correction ● 4 – Off-Book ● 2 – Outside Spread ● 1 – Off-Hours	Bitfield
36	UTC time of the trade	Time (up to nanoseconds)
38	Executed side	Numeric (0 – bid side, 1 ask side)
60	Trade buyer	String

61	Trade seller	String
71	Trade type as reported by the source	String
72	Time of when the trade was canceled	Time (up to nanoseconds)
201	Trade yield	Numeric

The trade message will look something like this:
trade: *{"instruments":[{"requestid":3,"insref":2926,"trade":[{"3":"2018-04-27","12":"147.84","13":"85","14":"584266","15":"224","21":"HM B","22":"H&M B","36":"14:34:44.806958385","60":"AVA","61":"AVA"}]}]}*

order:

The order array can have 6 different arrays in it, insert, delete and update for both bid and ask side

Array name	**Description**
7	Insert level on bid side
8	Insert level on ask side
9	Delete level on bid side
10	Delete level on ask side
11	Update level on bid side
12	Update level on ask side

The bidside arrays can have the following properties in it:

Property	**Description**	**Format**
3	Date of the update	Date
4	Time of the update	Time (up to nanoseconds)
5	Bid price on the current level	Numeric
16	Current level of the orderbook	Numeric (mandatory)
17	Number of bid orders on the current level	Numeric
19	Bid quantity on the current level	Numeric
62	Identifies the counter-parties interested in buying the instrument on the current level in the order book. The counter-parties are sorted from right to left in priority order. The list of counter-parties will usually be only the top 4 and not all the counter-parties	String

The askside arrays can have the following properties in it:

Property	**Description**	**Format**
3	Date of the update	Date
4	Time of the update	Time (up to nanoseconds)
6	Ask price on the current level	Numeric
16	Current level of the orderbook	Numeric (mandatory)
18	Number of ask orders on the current level	Numeric

20	Ask quantity on the current level	Numeric
63	Identifies the counter-parties interested in buying the instrument on the current level in the order book. The counter-parties are sorted from right to left in priority order. The list of counter-parties will usually be only the top 4 and not all the counter-parties	String

An example of an image when requesting the orderbook for insref 772 will look like this:

{"instruments":[{"insref":772,"requestid":4,"order":[{"7":[{"3":"2018-03-02","4":"14:30:44.224","5":"65.60","16":1,"17":"2","19":"3115","62":null},{"3":"2018-03-02","4":"14:30:44.224","5":"65.58","16":2,"17":"8","19":"11751","62":null},{"3":"2018-03-02","4":"14:30:19.529","5":"65.56","16":3,"17":"5","19":"6151","62":null},{"3":"2018-03-02","4":"14:30:44.224","5":"65.54","16":4,"17":"4","19":"3155","62":null},{"3":"2018-03-02","4":"14:30:44.593","5":"65.52","16":5,"17":"5","19":"4118","62":null}],"8":[{"3":"2018-03-02","4":"14:30:36.282","6":"65.62","16":1,"18":"3","20":"4638","63":null},{"3":"2018-03-02","4":"14:30:34.854","6":"65.64","16":2,"18":"7","20":"8153","63":null},{"3":"2018-03-02","4":"14:30:41.169","6":"65.66","16":3,"18":"9","20":"10820","63":null},{"3":"2018-03-02","4":"14:30:32.196","6":"65.68","16":4,"18":"8","20":"12205","63":null},{"3":"2018-03-02","4":"14:30:15.086","6":"65.70","16":5,"18":"9","20":"10507","63":null}]}]}]}

An update will look like this:

{"instruments":[{"insref":772,"requestid":4 ,"order":[{"7":[{"3":"2018-03-02","4":"14:30:44.224","5":"65.60","16":1,"17":"2","19":"3115","62":null}]}]}]}

performance:

The performance array can have the following properties in it:

Property	Description	Format
1024	diff1d	Numeric
1025	diff1dprc	Numeric
1026	diffyield1d	Numeric
1027	diffyield1dprc	Numeric
1028	diff1w	Numeric
1029	diff1wprc	Numeric
1030	diff2w	Numeric
1031	diff2wprc	Numeric
1032	diff1m	Numeric
1033	diff1mprc	Numeric
1034	diff3m	Numeric
1035	diff3mprc	Numeric
1036	diff6m	Numeric
1037	diff6mprc	Numeric
1038	diff9m	Numeric
1039	diff9mprc	Numeric
1040	diff1y	Numeric
1041	diff1yprc	Numeric
1042	diff3y	Numeric

1043	diff3yprc	Numeric
1044	diff5y	Numeric
1045	diff5yprc	Numeric
1046	diffytd	Numeric
1047	diffytdprc	Numeric
1048	diff10y	Numeric
1049	diff10yprc	Numeric

An update will look like this:
{ "instruments":[{"requestid" :"1", "insref": 354,"performance" :[{"1024":"4.600000","1025":"0.977476","1026":null,"1027":null,"1028":"12.500000","1029":"2.701534","1030":"35.800000","1031":"8.147474","1032":"86.400000","1033":"22.222222","1034":"-57.600000","1035":"-10.810811","1036":"-3.400000","1037":"-0.710405","1038":"20.400000","1039":"4.485488","1040":"39.800000","1041":"9.141020","1042":"90.900000","1043":"23.653396","1044":"126.000000","1045":"36.082474","1048":"336.400000","1049":"242.363112","1046":"-36.000000","1047":"-7.042254"}]}]}

netorderimbalance:

The netorderimbalance array name is shortened to noii and can have the following properties in it:

Property	**Description**	**Format**
3	Date of the update	Date
4	UTC time of the update	Time
5	Bid price	Numeric
6	Ask price	Numeric
7	Last price	Numeric
10	Quantity	Numeric
19	Bid quantity	Numeric
20	Ask quantity	Numeric
343	Imbalance	Numeric
344	Imbalance direction. Valid values: ● 1 – Buy Imbalance ● 2 – Sell Imbalance ● 3 – No Imbalance ● 4 – Insufficient Orders to Calculate	Numeric
345	Crosstype	Numeric

An update will look something like this:
{ "instruments":[{"insref": 104,"requestid": "1","noii": [{“5”:”12”,”6”:”13”,”343”:”300”,”344”:”1”}]}]}

Changelog

2020-04-15 *Added marketplace, instrumenttype, instrumentsubtype and unsubscribe as requestparameters*

Changed requestid to numeric, strings still valid but is deprecated

Changed requests parameter to request, requests is still valid but deprecated

Removed message newstext

Added messagetype Performance

Added messagetype Netorderimbalance

Added Bid quantity and Ask quantity for quote message

Added array name in the messagetype table

Changed the unsubscribe behavior, it is now only sent with the requestid.

Added entitlementpackage, newscoding, newcodingtags,
newscodingtype, newscodingmarketplace, newscodinglist and language in newsheadline

Added fieldtype specification, andded types bitfield Tabular String, Bitfield and Array Striing

2020-04-21 *Changed newcodingtype to numeric*

Added field Name source to newsheadlines

2020-06-18 *Changed id in request to mandatory*

Added datatype uint

Changed client message with type numeric to uint