Nordnet External API Feeds

After successful HTTP login the client should connect to one public feed to be able to subscribe to public market info and one private feed to be able to get own orders and trade changes in real time.

The client must always be able to handle the load created by the feed. The load in the private feed is low but when using the public feed the client should not subscribe to more instrument than the client can handle. If the client fails to read from the feeds in time the server will disconnect the client.

The client must be able to handle fields not covered in the documentation. Nordnet can at any time add fields to the feed messages, fields not documented should be considered as features and can be removed at any time.

The client must be able to handle feed disconnects during trading hours. When reconnects fail the clients must implement some sort of back off scheme so the client does not reconnect in a tight loop.

Private order and trade feed

Introduction

Private orders and trades are trades and orders belonging to the connecting session.

When the client has connected to the feed the client must send a login request. If the login fails the client will be disconnected. If the login is correct the feed will start send as soon there is data to send.

The private order and trade notifications are sent as JSON messages. The field "type" is used as indicator if this is an order or a trade. Other message types can be added later so the client must be able to handle unexpected JSON messages and not documented fields. The client should never have more than one private feed connection open at any time. A heartbeat is sent from the server every fifth second.

All communication on the feed is using SSL.

All messages sent to the private feed must be ended by newline.

Host and port to the feed is received in the HTTP login reply see login.

Login request

Connect to the server with a socket login with the session cookie. The session cookie is used for identification.

Login request format:

Key	Value
"cmd"	login
"args"	Arguments (see below)

Arguments:

Key	Mandatory	Description
"session_key"	true	The session key from the HTTP login.

Example of login request

Login and not get the current state of orders and trades.

```
{"cmd":"login", "args":{ "session_key":"ABC123"}}
```

Private Feed messages

Feed messages are initiated from the server and will be sent as soon as an order or trade event is happening.

Heartbeat

Heartbeats are sent when the feed has been silent for 5 seconds.

Key	Value
"type"	"heartbeat"
"data"	empty hash

Order notifications

The order messages is sent from the server when an order belonging to one of the end users accounts is changed. Sometimes the server only send order_id and fields that has changed since the last message of that order. So the client must be able to handle not complete order messages. The order message fields are the same as the http request GET orders reply.

Key	Value
"type"	"order"
"data"	See HTTP GET orders request

Trade notifications

The trade messages is sent from the server when an order belonging to one of the end users is traded. The trade message fields are the same as the http request GET trades reply.

Key	Value
"type"	"trade"
"data"	See HTTP GET trades request

The public trade and price feed

The feed service sends the public trades and price information to the connected clients. The client can subscribe to six different message types. Each subscription is related to one entity (one instrument or one news source). The client should never have more than one public feed connection open at any time. A heartbeat is sent from the server every fifth second. The client can subscribe to multiple entities and message types at the same time.

The message types *NOTE*: Not all feeds are available from all markets:

- Trade - all trade on the market.
- Price - last, best bid, best ask and so on.
- Depth - the orderbook price depth in five levels.
- Trading status - the state of an orderbook (e.g continuous trading, stopped, etc).
- Indicator - price information of indicators.
- News - news in realtime (if the user has the right to see it)

All communication on the feed is using JSON.

The application must use the login command before any subscriptions can be requested. If the login fails or a subscription is made before the login the connection will be closed.

If the client application can’t read from the feed in time and the feed starts to buffer the connection will be closed.

The client must be able to handle not documented fields and messages.

A heartbeat is sent from the server when no other data has been sent for 5 seconds.

Host and port to the feed is received in the HTTP login reply.

All requests must end with linefeed (ASCII 10). The requests are on JSON format.

In price, depth and indicator messages only changed fields and identification of the entity is sent. If an optional field like for example ep is removed the value null is sent in the field.

Ticks and conflation

The conflation flag is a bool flag, which we conflate the subscription. If set to true you will get at most one update per second. The default is conflation set to false.

There are two types of ticks: delta and full. The full tick is the full JSON object with every field. However, the delta tick is *only* the object with the fields who have been updated since the last tick. Please note that if you are using the conflation-flag you will always get full tick.

Login request

Connect to the server with a socket login with the session cookie. The session cookie is used for identification.

Login request format:

Key	Value
"cmd"	login
"session_key"	The session key from the HTTP login.

Example of login request

```
{"cmd":"login", "args":{ "session_key":"ABC123"}}
```

Subscribe/unsubscribe requests

Each subscribe/unsubscribe message is related to one entity at the time. Multiple subscriptions is allowed on one public feed.

Key	Value
"cmd"	"subscribe" or "unsubscribe"
"session_key"	The session key from the HTTP login.
"args"	see below

Args: |Key |Value | |-------------|----| |"t" |Type - "price", "depth", "trade", "indicator", "news" or "trading_status".| |"i" |The tradable identifier or indicator identifier. Used for "price", "depth", "trade", "indicator" and "trading_status".| |"m" |The tradable market_id or indicator src. If . Used for "price", "depth", "trade", "indicator" and "trading_status". If this is used for "indicator" the field is a string otherwise an integer.| |"s" |News source. Only used for "news"| |"delay" |(Deprecated) True if delayed news are accepted. Default is false. Used only for "news"|

Examples of subscribe requests

Subscribe to Nordnet B (identifier 1869) on Burgundy (marketplace 30).

Price tick:

```
{"cmd":"subscribe", "args":{"t":"price", "i":"1869", "m":30}}
```

Order Depth:

```
{"cmd":"subscribe", "args":{"t":"depth", "i":"1869", "m":30}}
```

Trades:

```
{"cmd":"subscribe", "args":{"t":"trade", "i":"1869", "m":30}}
```

Trading status:

```
{"cmd":"subscribe", "args":{"t":"trading_status", "i":"1869", "m":30}}
```

Index:

```
{"cmd":"subscribe", "args":{"t":"indicator", "i":"SIX-IDX-DJI", "m":"SIX"}}
```

News:

```
{"cmd":"subscribe", "args":{"t":"news", "s":2}}
```

Feed messages

The messages sent from the server has the following structure

Key	Value
"type"	"price", "depth", "trade", "indicator", "news", "trading_status", "heartbeat"

Key	Value
"data"	Depends on type

Price Feed

In the first message after a subscription all available fields are sent to the client. After that only "m","i" and changed fields are sent. All fields are not available on all markets.

Key	Value
"i"	The identifier of the tradable.
"m"	The market_id of the tradable.
"trade_timestamp"	Timestamp of the last trade - the time when last and turnover fields are updated.
"tick_timestamp"	The timestamp of this update.
"bid"	Best bid price.

Key	Value
"bid_volume"	Total volume of the best bid price.
"ask"	Best ask price.
"ask_volume"	Total volume of the best ask price.
"close"	Closing price.
"high"	Highest price for a trade today.
"last"	Last paid.
"last_volume"	Volume of the last trade.
"low"	Lowest price for a trade today.

Key	Value
"open"	Opening price.
"turnover"	Turnover value today. The currency is the orderbook currency.
"turnover_volume"	Turnover volume today.
"ep"	Equilibrium price. Only sent during auction on the Nordic markets.
"paired"	The paired quantity of shares at the equilibrium price. Only sent during auction on the Nordic markets.
"imbalance"	The quantity of shares not paired at the equilibrium price. Positive quantity indicates buy imbalance. Negative quantity indicates sell imbalance. Only sent during auction on the Nordic markets.

Trade Feed

The trade feed contains public trade information.

Key	Value

Key	Value
"i"	The identifier of the tradable.
"m"	The market_id of the tradable.
"trade_timestamp"	The time of the trade.
"price"	The price of the trade. Currency is the default currency of the orderbook.
"volume"	Volume traded.
"broker_buying"	Buying participant. Not supported on all markets.
"broker_selling"	Selling participant. Not supported on all markets.
"trade_type"	The exchange trade id. Not supported on all markets.

Key	Value
"trade_type"	The exchange trade type - different on different markets.

Order depth feed

The first message after a subscription all available fields are sent to the client. After that only "m","i" and changed fields are sent.

Key	Value
"i"	The identifier of the tradable.
"m"	The market_id of the tradable.
"tick_timestamp"	The timestamp of this update.
"bid1"	Bid price on level 1
"bid_volume1"	Bid volume on level 1

Key	Value
"ask1"	Ask price on level 1
"ask_volume1"	Ask volume on level 1
"bid2"	Bid price on level 2
"bid_volume2"	Bid volume on level 2
"ask2"	Ask price on level 2
"ask_volume2"	Ask volume on level 2
"bid3"	Bid price on level 3
"bid_volume3"	Bid volume on level 3

Key	Value
"ask3"	Ask price on level 3
"ask_volume3"	Ask volume on level 3
"bid4"	Bid price on level 4
"bid_volume4"	Bid volume on level 4
"ask4"	Ask price on level 4
"ask_volume4"	Ask volume on level 4
"bid5"	Bid price on level 5
"bid_volume5"	Bid volume on level 5

Key	Value
"ask5"	Ask price on level 5
"ask_volume5"	Ask volume on level 5

Trading status feed

Key	Value
"i"	The identifier of the tradable.
"m"	The market_id of the tradable.
"tick_timestamp"	Timestamp of the update

Key	Value
"status"	The Nordnet trading status of an orderbook. • "C" - Continuous trading. • "R" - Receiving orders but trading is disabled. • "D" - Temporarily disabled. • "X" - Permanently disabled. • "U" - Unknown state.
"source_status"	The marketplace orderbook state.
"halted"	The halted reason if available from the market.

Indicator feed

Key	Value
"i"	Index identification.
"m"	Index source.

Key	Value
"tick_timestamp"	Timestamp of the last update
"high"	Highest price today.
"low"	Lowest price today.
"last"	Last price.
"close"	Last closeprice.

News feed

Key	Value
"item_id"	News itemid can be used in the news rest service to get the whole news item.
"lang"	The language of the news (this is set by the source)

Key	Value
"datetime"	The time of the news.
"source_id"	News source id.
"headline"	Headline of the news
"instruments"	Array of instruments involved in the news. This is only set if the news source provides the information.

Examples of feeds

Price feed:

```
{"cmd":"subscribe", "args":{"t":"price", "m":78, "i":"F00000X27D"}}

{"type":"price","data":{"delayed":900,"i":"F00000X27D",
"m":"SIX","tick_timestamp":1560168002590,"high":0.50,"low":1.00,"close":213.94,"last":253.94}}
```

Depth feed:

```
{"cmd":"subscribe", "args":{"t":"depth", "i":"1869", "m":30}}

{"type":"depth","data":{"i": "1869", "m":30, "tick_timestamp":1560168002590,
"bid1":21, "bid_volume1":100, "ask1": 35, "ask_volume1":300}}
```

Index:

```
{"cmd":"subscribe", "args":{"t":"indicator", "i":"SIX-IDX-DJI", "m":"SIX"}}

{"type":"indicator","data":{"delayed":900,"i":"SIX-IDX-DJI",
"m":"SIX","tick_timestamp":1560168002590,"high":0.00,"low":0.00,"close":25983.94,"last":25983.94}}
```

News:

```
{"cmd":"subscribe", "args":{"t":"news", "s":1}}

{"type":"news","data":{"news_id":1000729440,"source_id":1,
"headline":"IDL BIOTECH: HOPPAS KUNNA BEHALLA TILLVAXTTAKT PÅ 20% - VD",
"lang":"sv","type":"NEWS","timestamp":1560173163000,"instruments":[17513784]}}
```

Last updated 2020-12-10 17:23:34 +0100