



BMLL Data Feed

Level 2 quotes

Date: 2023-12-27

1.1. Introduction

The Level 2 quotes dataset offers a tick-by-tick view of the order book, aggregated by price level with up to 10 levels of depth. It provides an in-depth understanding of historical market activity, enabling users to make more informed trading decisions.

1.2. Dataset schema

Basic Types

Type	Content	Text representation
bigint	8 byte signed integer	103403403403
char(n)	fixed length string containing letters, digits or '@'	XLON
date	Calendar date	YYYY-MM-DD
enum	string from a defined set	AUCTION
integer	4 byte signed integer	1234
price	8 byte binary floating point. All prices are rounded to a fixed number of decimal places. Trailing decimal zeros are not shown.	123.45
quantity	8 byte binary floating point, rounded to 4 decimal places	125
timestamp	Nanoseconds since 1970 in UTC. Timestamp will be represented in ISO format in text format.	20220302-11:23.5712345789
varchar	variable length string up to 500 characters. May contain quotes and special characters. All text will be utf8-encoded unless indicated differently. Ticker and ISO codes should be ASCII compatible.	"Vodafone Group"
bool	Boolean	1, 0

1.3. Schema

One row in the dataset corresponds to a single, atomic event during the trading day.

Field Name	Type	Description
Ticker	varchar (ascii)	FactSet ticker. It refers to the unique identifier used within the FactSet financial data platform to represent a specific security.
ISOExchangeCode	varchar (ascii)	The ISO code of the exchange.
TradeDate	date	The specific date on which a trade was executed - the date is derived from the LocalTimestamp.
EventTimestamp	timestamp	The UTC timestamp indicates the date and time at which the update occurred in UTC timezone, with microsecond precision.
LocalTimestamp	timestamp	The local timestamp indicates the date and time at which the update occurred in the timezone of the primary record associated with the venue, with microsecond precision.
TimestampNanoseconds	bigint	This column contains a representation of the time of the update as the number of nanoseconds since 1970-01-01 00:00:00.0 UTC.
TZOffset	int	The offset between the local timestamp and UTC in seconds. Note: This is based on the original listing location of the primary instrument. Format was +/- HHMM . This will be adjusted during the DST if such a change occurs.
Core MBP fields (Price levels 1-10 are available)		
BidPrice[1-10]	price	The price of the price level on the bid side in local currency
BidQuantity[1-10]	quantity	The number of shares submitted at the price level on the bid side
BidNumOrders[1-10]	int	The number of distinct orders at given the price level on the bid side
AskPrice[1-10]	price	The price of the price level on the ask side in local currency
AskQuantity[1-10]	quantity	The number of shares submitted at the price level on the ask side
AskNumOrders[1-10]	bigint	The number of distinct orders at given the price level on the ask side
Additional fields		
BidLevelCount	int	Number of bid levels populated
AskLevelCount	int	Number of ask levels populated

L2EventNo	bigint	L2EventNo is a data field that enables to sort rows of financial data that occur at the same timestamp. The first event of a particular day or specific event is labelled as 1 in the L2EventNo field. L2EventNo increases then with every update.
MIC	varchar	The Market Identifier Code (MIC) is used to uniquely identify the exchange for which the L2-book is reconstructed.
ExchangeTicker	varchar	The original ticker as provided by the exchange.
BMLLObjectId	bigint	The BMLL Listing Id for the instrument
CurrencyCode	char(3) (ascii)	The currency of the order book information; as provided by the exchange.
MarketState	enum	The market state, normalised by BMLL.
ExchangeSequenceNo	bigint	The sequence number of the market state update, if provided by the exchange.
BMLLSequenceNo	bigint	A synthetic sequence number as provided by BMLL. This ensures the correct ordering of all messages from a particular source and should be used as the standard to sort and join all messages.
BMLLSequenceSource	bigint	The source of the sequence. Used if there is more than one feed, for example where there are off book and on book trade messages provided by different feeds.

1.4. Market State Normalisation

BMLL provides a consistent normalisation for market state messages, in order to make cross venue analysis quick and easy. For full details, please refer to the BMLL Data Feed: Market State documentation.

1.5. File format and delivery structure

Data is delivered as a single file per market per date, as a parquet file.

1.6. Data arrival times and coverage

Full product coverage and arrival times are available at <https://data.bmlitech.com>.