INDEX Protocol

Chain structure: blocks, transactions, data

Green Paper, v0.1

14/11/2018

Market metrics

- Our forecast (up to 2023)
  - Trading assets: 10 000 (4 000 at Q4’2018)
  - Assets, used as currency: 2 000 (~250 at Q4’2018)
  - Avg. trading activity: 7B trades per Year (2.8B at 2018)
  - Avg. trades per one trading pair: 100M per Year (~30M at 2018)
    - Up to 3000 trades per 15 min time frame at 2018

Data block

- Собираются все данные с источников (транзакции с бирж, etc)
- Группируются по биржам и по валютным парам
- Фрейм – это группа транзакций с одной биржи и по одной паре
- Внутри фрейма – сортируются по времени
- Каждый отдельный фрейм содержит заголовок и подпись
- Все фреймы от одной биржи

Base types: Exchange Trade

- **64 bit** – unique trade ID from exchange.
  - *We NOT connected any exchange without trade ID available over API*
- **32 bit** – trading pair id (*point of optimization*)
- **32 bit** – trader ID (reserved for a future)
- **64 bit** – trade datetime in UTC
  - *Exchange will provide any type of this data – ISO-format, UTC, UTC with ms. and any type of time zone (local times or UTC). We convert all to one format*
- ~~**64 bit** – UTC loading datetime (*point of optimization*)~~
- **64 bit** – trading price (as unsigned float), as reported from exchange (*point of optimization*)
- **64 bit** – trading amount as reported from exchange (*point of optimization*)
  - *Some reported full amount of orders, without or including fee. Its only filled amount of order*
- **1 bit** – SELL/BUY side (*point of optimization*)

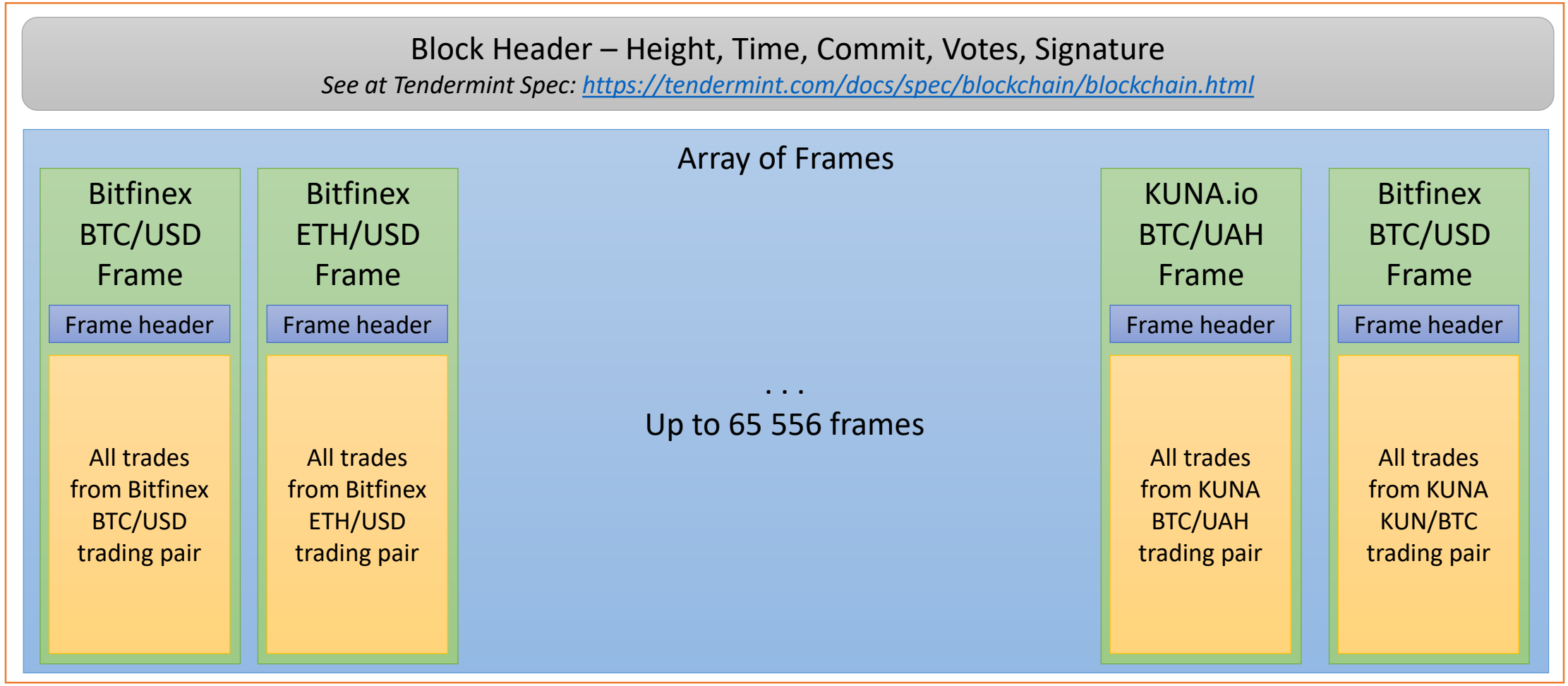
In total: **385 bits** (without optimization). Possible to packet up to **289 bit or less**

Base type: Frame header

- Frame – container for all trades from one source at same period
  - **16 bit** – source ID, e.g. exchange, data provider
  - **64 bit** – packed data about asset
    - 16 bit – asset ID
    - 16 bit – currency ID
    - 8 bit – type of market (SPOT, T+1, Future, Option, Swap)
    - 2 bit – type of API or connection
    - 6 bit – counter of used nodes to obtain quotes
    - 16 bit – reserved field
  - **64 bit** - nonce of packet (incremental from previously)
  - **512 bit** - ED25519 signature of frame (exchange private key signature)
  - **32 bit** – 16 bit as transaction counter (and 16 bit reserves to future)
  - **256 bit** – SHA-256 previously frame (not a block, only frame)
  - **64 bit** – UTC timestamp, beginning time of block
  - ~~**32 bit** – duration of period, in milliseconds~~ (*point of optimization, e.g. fixed by protocol spec to 900 sec.*)
  - **256 bit** – SHA-256 hash of TradesBlock
  - **16 bit** – length of one transaction at TradesBlock field (count bits per trade)
  - TradesBlock – array of bits, up to 18939615 (max 65556 transactions with 289 b/trx)

Header: 1 280 bit (160 bytes, without duration field) and trades block, avg. now 5K trx – 1 445K bits.
Avg total size: 1 572 Kbytes

Block structure



Open questions for R&D

- Compress data (exchanges transactions)
  - LZMA2
  - Brotli
  - Any other algo??
- Store compressed data inside a DataBlock
- Binary format+compress vs compact-json+compress
- What to compress?
  - Each block (trading pair)
  - All from one exchange
  - All from frame

Test data set

- One exchange – Binance
- All trading pairs – 260+
- 15 min time frame, randomly 100 frames (from 11/02 to 20/11’2018)
- Size: 240 Mb
- Testing algorithms:
  - Deflate
  - Brotli
  - LZ4 and LZ5
  - LZMA2
  - Zstandard

Results

- After binary pack (using PHP pack func.) one trade: 37 byte.
- Test samples:
  - Min: 12 trades per 15 min, 444 bytes
  - Max: 991 trades, 36 667 bytes
- Best result: **LZMA2**
  - 12 trades/444 bytes compressed to 192 bytes (**2.31 ratio**)
  - 991 trades/36 667 bytes compressed to 11 472 bytes (**3.19 ratio**)