



Simple Binary Encoding

Version 2.0 RC 1

Technical Proposal

August 2018

v0.1

Proposal Status: Draft

For Global Technical Committee Governance Internal Use Only

Submission Date		Control Number	
Submission Status		Ratified Date	
Primary Contact Person		Release Identifier	

DISCLAIMER

THE INFORMATION CONTAINED HEREIN AND THE FINANCIAL INFORMATION EXCHANGE PROTOCOL (COLLECTIVELY, THE "FIX PROTOCOL") ARE PROVIDED "AS IS" AND NO PERSON OR ENTITY ASSOCIATED WITH THE FIX PROTOCOL MAKES ANY REPRESENTATION OR WARRANTY, EXPRESS OR IMPLIED, AS TO THE FIX PROTOCOL (OR THE RESULTS TO BE OBTAINED BY THE USE THEREOF) OR ANY OTHER MATTER AND EACH SUCH PERSON AND ENTITY SPECIFICALLY DISCLAIMS ANY WARRANTY OF ORIGINALITY, ACCURACY, COMPLETENESS, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. SUCH PERSONS AND ENTITIES DO NOT WARRANT THAT THE FIX PROTOCOL WILL CONFORM TO ANY DESCRIPTION THEREOF OR BE FREE OF ERRORS. THE ENTIRE RISK OF ANY USE OF THE FIX PROTOCOL IS ASSUMED BY THE USER.

NO PERSON OR ENTITY ASSOCIATED WITH THE FIX PROTOCOL SHALL HAVE ANY LIABILITY FOR DAMAGES OF ANY KIND ARISING IN ANY MANNER OUT OF OR IN CONNECTION WITH ANY USER'S USE OF (OR ANY INABILITY TO USE) THE FIX PROTOCOL, WHETHER DIRECT, INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL (INCLUDING, WITHOUT LIMITATION, LOSS OF DATA, LOSS OF USE, CLAIMS OF THIRD PARTIES OR LOST PROFITS OR REVENUES OR OTHER ECONOMIC LOSS), WHETHER IN TORT (INCLUDING NEGLIGENCE AND STRICT LIABILITY), CONTRACT OR OTHERWISE, WHETHER OR NOT ANY SUCH PERSON OR ENTITY HAS BEEN ADVISED OF, OR OTHERWISE MIGHT HAVE ANTICIPATED THE POSSIBILITY OF, SUCH DAMAGES.

DRAFT OR NOT RATIFIED PROPOSALS (REFER TO PROPOSAL STATUS AND/OR SUBMISSION STATUS ON COVER PAGE) ARE PROVIDED "AS IS" TO INTERESTED PARTIES FOR DISCUSSION ONLY. PARTIES THAT CHOOSE TO IMPLEMENT THIS DRAFT PROPOSAL DO SO AT THEIR OWN RISK. IT IS A DRAFT DOCUMENT AND MAY BE UPDATED, REPLACED, OR MADE OBSOLETE BY OTHER DOCUMENTS AT ANY TIME. THE FPL GLOBAL TECHNICAL COMMITTEE WILL NOT ALLOW EARLY IMPLEMENTATION TO CONSTRAIN ITS ABILITY TO MAKE CHANGES TO THIS SPECIFICATION PRIOR TO FINAL RELEASE. IT IS INAPPROPRIATE TO USE FPL WORKING DRAFTS AS REFERENCE MATERIAL OR TO CITE THEM AS OTHER THAN "WORKS IN PROGRESS". THE FPL GLOBAL TECHNICAL COMMITTEE WILL ISSUE, UPON COMPLETION OF REVIEW AND RATIFICATION, AN OFFICIAL STATUS ("APPROVED") OF/FOR THE PROPOSAL AND A RELEASE NUMBER.

No proprietary or ownership interest of any kind is granted with respect to the FIX Protocol (or any rights therein).

Copyright 2003-2018 FIX Protocol Limited, all rights reserved.

Table of Contents

Contents

Table of Contents	3
Document History	4
1 Introduction	5
1.1 Authors.....	5
2 Requirements.....	6
2.1 Business Requirements.....	6
2.1.1 Enhanced template extension	6
2.2 Technical Requirements.....	6
2.2.1 Specification clarifications	6
2.2.2 Alignment of blocks.....	6
3 Issues and Discussion Points	6
3.1 Resolved Issues	6
3.2 Compatibility.....	7
3.3 Usage for non-FIX semantics.....	7
4 References	7
5 Relevant and Related Standards	8
6 Intellectual Property Disclosure.....	8
7 Definitions.....	8
8 Simple Binary Encoding.....	9
8.1 Specifications	9
8.2 Schema.....	9
Appendix B – Compliance Strategy	9
XML Validation	9
Compliance Test Suite.....	9

1 Introduction

The High Performance Working Group was formed with the goal of improving the fit-for-purposefulness of FIX for high performance message exchange. The working group has developed multiple standards for the presentation layer (message encoding), including Simple Binary Encoding (SBE). SBE is optimized for ultra-low latency encoding and decoding of messages, orders of magnitude better than typical tag value message processors, and with far more deterministic performance.

SBE is distinguished by these characteristics:

- A binary type system that maps FIX datatypes to native platform types, avoiding unnecessary and costly translation between character-based encodings and types directly usable by computers.
- Deterministic and narrowly targeted message layouts enforced by templates. They enable each use case of an overloaded FIX message type such as ExecutionReport to have its own layout. Each template contains just the required fields for its use case, such as immediate execution. The benefit is reduced message size and reduced optionality. In hardware terms, the result is reduced network bandwidth, reduced memory usage, reduction of cache misses, and less code branching in CPUs. All add up to deterministic performance and low latency. (In fact, “mechanical sympathy” enables hardware-based or accelerated solutions.)
- Just data on the wire. Metadata is exchanged out-of-band as a message schema. This enables pre-processing such as code generation of encoders and decoders rather on-the-fly interpretation.
- An explicit versioning mechanism that allows extension of templates over time without breaking older decoders.

Version 1.0 of SBE was the first FIX standard to complete the full cycle of technical standard process. Due its benefits, SBE v1.0 has gained users both within and beyond the financial industry for low latency applications.

Nevertheless, the user community has requested some important enhancements and pointed out some ambiguities or deficiencies in the original standard. Therefore, we are proposing version 2.0. This effort is Release Candidate 1 of that new version.

1.1 Authors

Name	Affiliation	Contact	Role
Don Mendelson	Silver Flash LLC	Donmendelson@silver-flash.net	SBE lead

2 Requirements

2.1 Business Requirements

2.1.1 Enhanced template extension

SBE has an explicit versioning mechanism that allows message templates to be extended over time without breaking older decoders. The benefit is that message definition changes may be migrated gradually without requiring a big-bang approach where ever client and server needs to update at the same time. In SBE version 1.0, message extension allowed new fields to be added to the end of a message block without breaking older decoders. It did not allow, however, adding of whole new repeating groups or variable-length data elements during template extension. It could only be achieved by distributing a new template. Several users commented that this restriction was unacceptable.

Therefore, in this release candidate, the template extension mechanism was enhanced to allow adding of repeating groups and variable-length data elements to both the root of a message as well as nested within a repeating group. To implement the change, two fields were added to the message headers, giving the count of repeating groups and count of variable-length data elements. Unfortunately, this renders the new header incompatible with version 1.0. Since they are not-directly interoperable, it was decided to call this release version 2.0. However, converting a version 1.0 message schema to version 2.0 is a straightforward process.

2.2 Technical Requirements

2.2.1 Specification clarifications

Most of the issues that were entered were for document clarifications rather than wire format enhancements.

2.2.2 Alignment of blocks

In version 1.0, a mechanism to control the byte alignment of fields was provided (as well as byte order, or endianness). The usual motivation is to make fields align on memory boundaries to improve CPU performance (another “mechanical sympathy”). However, it was awkward or impossible to make variable-length data or instances of repeating group align properly. Therefore, an alignment attribute was added for that purpose.

3 Issues and Discussion Points

3.1 Resolved Issues

Issues were tracked in GitHub. These issues were resolved and accepted for version 2.0 Release Candidate 1. See [issues](#) and [pull requests](#) in GitHub for details and changes.

Issue	Description
26	Adding new variable length field in a repeating group
31	Presence attribute belongs to a field
35	Field attributes
36	Document cross-references
37	Extend message with repeating groups and vardata
39	sinceVersion on nested types within a composite
40	Clarification on constant char encodings
44	Remove deprecated attributes
45	Update examples for changed headers
48	Member names for data encoding type
51	Semantic type with Decimal encoding
59	refType is missing semanticAttributes
62	Decimal representation for optional mantissa with non-constant exponent
64	Support for easy constant value setting in fields
65	Alignment of var data and repeating groups
87	update examples for version 2.0

3.2 Compatibility

Version 2.0 is not interoperable with SBE version 1.0, either in wire format or XML schema. However, converting an existing version 1.0 message schema to version 2.0 is straightforward.

3.3 Usage for non-FIX semantics

SBE is gaining popularity beyond FIX. However, it only describes the mapping of FIX datatypes. We can anticipate that users will want to map other common datatypes for their applications, e.g. IP address, URL, and so forth. SBE is highly flexible as it is, but there may be requests to generalize it beyond its original uses, even outside the financial industry.

4 References

Reference	Version	Relevance	Normative
FIX Simple Binary Encoding Technical Specification	RC1		Yes

GitHub project FIXTradingCommunity/fix-simple-binary-encoding		Final specifications as well as working drafts and issue tracking.	

5 Relevant and Related Standards

Related Standard	Version	Reference location	Relationship	Normative
None				

6 Intellectual Property Disclosure

Related Intellection Property	Type of IP (copyright, patent)	IP Owner	Relationship to proposed standard
None			

7 Definitions

Term	Definition

8 Simple Binary Encoding

8.1 Specifications

Full specifications for the Simple Binary Encoding are available in separate document (*FIX Simple Binary Encoding Technical Specification v2.0 RC1*). The standard defines wire format and message schema declaration. The document is a snapshot of drafts now being developed in GitHub project [FIXTradingCommunity/fix-simple-binary-encoding](https://github.com/FIXTradingCommunity/fix-simple-binary-encoding).

8.2 Schema

An XML schema (XSD) is provided to standardize XML message schemas. The XSD file is publicly available in GitHub project in GitHub project [FIXTradingCommunity/fix-simple-binary-encoding](https://github.com/FIXTradingCommunity/fix-simple-binary-encoding). The XML schema is also served by the address corresponding to its XML namespace, <http://fixprotocol.io/2017/sbe/>.

Appendix A - Usage Examples

Examples are provided in the specification document.

Appendix B – Compliance Strategy

XML Validation

Message schemas should be validated against the provided XML schema (XSD).

Compliance Test Suite

The FIX technical standard process requires that to be promoted to final specification, a draft standard must have at least two interoperable implementations. A compliance test suite was published publicly in GitHub for SBE version 1.0. By the time version SBE 2.0 reaches draft stage, this compliance test suite must be updated to version 2.0.