



FIX Global Technical Committee

Extension of BeginString for FIX Latest

October 28, 2021

Revision 0.4

Proposal Status: Approved

For Global Technical Committee Governance Internal Use Only

Submission Date	October 28, 2021	Control Number	EP270
Submission Status	Approved	Ratified Date	
Primary Contact Person	Hanno Klein, GTC	Release Identifier	FIX.Latest

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 2003-2022 FIX Protocol Limited, all rights reserved.

Table of Contents

Document History	5
1 Introduction	6
2 Business Requirements	6
2.1 Extension of BeginString(8)	7
2.2 Usage of ApplVerID(1128) and CstmApplVerID(1129)	8
3 Issues and Discussion Points	9
4 Proposed Message Flow	9
5 FIX Message Tables.....	9
6 FIX Component Blocks	9
7 Category Changes	9
Appendix A - Data Dictionary.....	10
Appendix B - Glossary Entries	11
Appendix C - Abbreviations	11
Appendix D - Usage Examples	11
Appendix E – Disposition of Public Comments	12

Table of Figures

Figure 1 - FIX Technical Standard Stack6

Document History

Revision	Date	Author	Revision Comments
0.1	September 27, 2021	Hanno Klein, FIX GTC	Initial draft
0.2	October 25, 2021	Hanno Klein, FIX GTC	Minor updates based on internal review
0.3	October 28, 2021	Hanno Klein, FIX GTC	Clarification of session profile names versus BeginString(8) values
0.4	December 21, 2021	GTC Technical Support	Included disposition of comments from Public Comment period.

1 Introduction

Extension Pack [EP260](#) added FIX Latest as an application version of FIX. Although technically also a version, FIX Latest emphasizes the concept of running a selected subset of FIX messages, components, fields, and values agreed by the counterparties over one of the four session profiles of the FIX Session Layer (see Figure 1 - FIX Technical Standard Stack). This extension proposal incorporates the session profile identifiers FIX.4.2 and FIX4 as values “FIX.4.2” and “FIX.4.4” to the BeginString(8) field in addition to the existing value “FIXT.1.1” representing the session profiles FIXT and LFIXT.

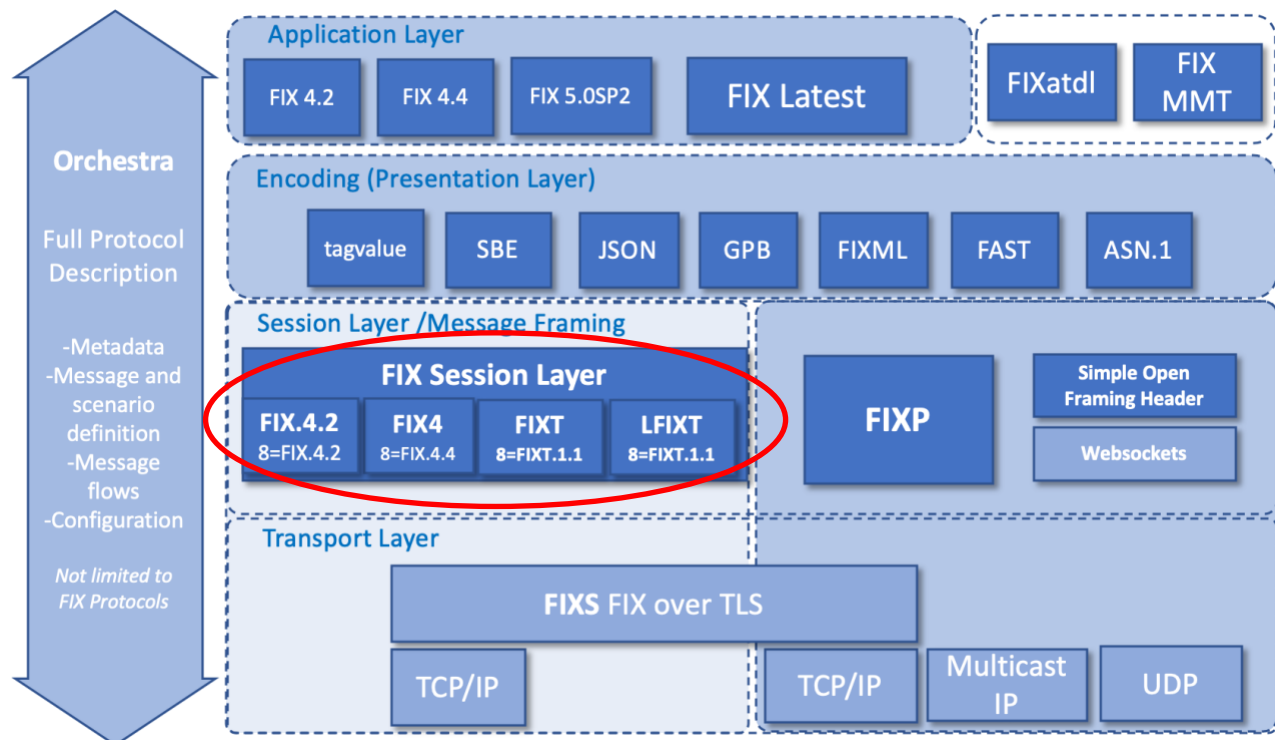


Figure 1 - FIX Technical Standard Stack

2 Business Requirements

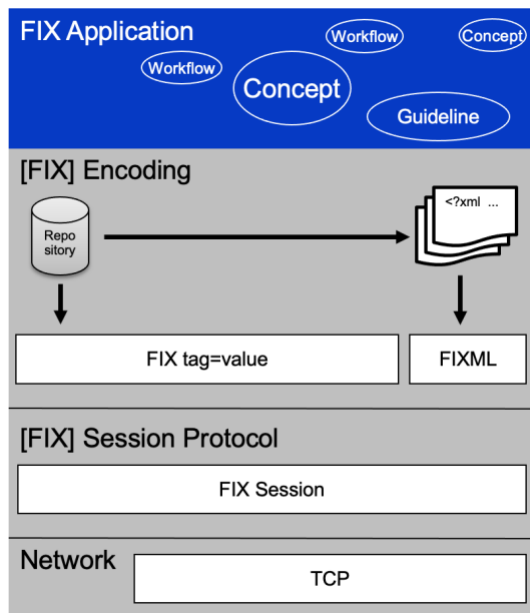
Counterparties have typically agreed on a single FIX version as part of their Rules of Engagement defining the specific messages, fields and values supported between them. In terms of strict FIX compliance, this would exclude the usage of any elements from higher versions, including the usage of new enumerations for existing fields or new message types.

However, regulatory requirements have more or less forced the FIX community to include user-defined fields, values, or to use new standard elements from higher versions - i.e. the notion of a single “FIX version” no longer applied to the vast majority of FIX connections out there. The usage of elements from higher versions is an official [FIX policy](#): *The Global Technical Committee (GTC) Governance Board recommends that fields/tags and enumeration values from later versions of FIX which meet the business or implementation requirements of FIX be retro-fitted into the implementation.*

The business requirement is to support the use of FIX session protocols other than FIX Transport (FIXT) that was introduced with FIX 5.0, especially to support the FIX session protocols for FIX 4.2 and FIX 4.4. This removes the barrier of having to change the FIX Session Layer in order to fully benefit from FIX Latest, i.e. to change the value of BeginString(8).

2.1 Extension of BeginString(8)

Initially, FIX did not offer any choices beneath the application layer in the technical stack. Up to and including FIX 4.2, the encoding had to be tag=value. Prior to FIX 5.0, the application and session layer were tightly coupled with a single value of BeginString(8) identifying both of them. The technical stack of FIX generation 4 (FIX 4.x) was monolithic with respect to the FIX Session Layer.



FIX 5.0 changed this paradigm by introducing the concept of Transport Independence and made the session protocol as of FIX 5.0 have its own name (FIX Transport a.k.a. FIXT) and versioning (v1.1). The value of BeginString(8) no longer contained the application layer version. It was replaced with the identification of the session protocol, i.e. "FIXT.1.1".

As of FIX 5.0, the identification of the application version became optional with either ApplVerID(1128) used on each application message or DefaultAppVerID(1137) used once in the Logon(35=A) message and applying to all messages sent in that session. This has not changed with FIX Latest.

It is proposed to support multiple values for BeginString(8) when using FIX Latest over the FIX Session Layer. It is not proposed to support the use of different values of BeginString(8) in the same session, i.e. the counterparties need to agree on the single value used for all messages of a given session. This value only identifies the session profile and does not limit the use of any messages, components, fields or values from FIX Latest. The following table maps the FIX session profiles to their values in BeginString(8) (source: <https://www.fixtrading.org/standards/fix-session-layer-online#fix-session-profiles>).

FIX session profile	BeginString(8)	Description
FIX.4.2	FIX.4.2	The FIX session profile for use with the FIX 4.2 application layer.
FIX4	FIX.4.4	The FIX session profile backward compatible with FIX 4.4 recommended when counterparties will only be using a single application version during the FIX session, such as <i>FIX Latest</i> .
FIXT	FIXT.1.1	The FIX session profile that must be used when mixing multiple application versions over the same FIX session. May be used with a single application version of FIX such as <i>FIX Latest</i> .
LFIXT	FIXT.1.1	Lightweight FIXT restricted session layer message recovery ¹ to simplify the protocol while maintaining compatibility with FIXT when using LFIXT compatible model of operation.

It is proposed that BeginString(8) supports the following values:

- “FIX.4.2”, i.e. session profile FIX.4.2
- “FIX.4.4”, i.e. session profile FIX4
- “FIXT.1.1”, i.e. session profiles FIXT and LFIXT

2.2 Usage of ApplVerID(1128) and CstmAppVerID(1129)

It is possible to explicitly identify the standard application version by using ApplVerID(1128) and ApplExtID(1156) in the StandardHeader component. This does not mean that all messages, components, fields, and values of this version are supported. The supported subset still needs to be agreed between the counterparties.

Note that only the use of BeginString(8)=“FIXT.1.1” together with ApplVerID(1128) continues to support the use of different application versions within a single session. For example, NewOrderSingle(35=D) and ExecutionReport(35=8) messages may use ApplVerID(1128)=4 (FIX42) while TradeCaptureReport(35=AE) uses ApplVerID(1128)=9 (FIX50SP2).

However, it is recommended to rather exchange [FIX Orchestra](#) XML files out-of-band to define the supported messages, components, fields, and values.

The <repository> element of a FIX Orchestra XML file contains the attribute “version” that can be used to define the one's own version of the meta-data used for the given interface. This value can then be conveyed in the field CstmAppVerID(1129) to identify the correct meta-data file.

Linking CstmAppVerID(1129) to Orchestra meta-data file is also an option to support backward compatibility by allowing two different values (e.g. current and previous interface version) in application messages. The message recipient can then draw upon the correct Orchestra XML file to validate the message content. Furthermore, the Orchestra pedigree attributes (e.g. added, updated, deprecated) may be used to reflect the changes in your interface.

¹ The LFIXT session profile uses the same BeginString(8) value as the FIXT session profile. The use of LFIXT and its mode of operation must be agreed upon out-of-band by counterparty agreement.

3 Issues and Discussion Points

There are no issues or discussion points.

4 Proposed Message Flow

There are no changes to existing FIX message flows.

5 FIX Message Tables

There are no changes to existing FIX message tables.

6 FIX Component Blocks

There are no changes to existing FIX component blocks.

7 Category Changes

There are no changes to existing categories.

Appendix A - Data Dictionary

Tag	FieldName	Action	Datatype	Description	FIXML Abbreviation	Add to / Deprecate from Message type or Component block
8	BeginString	CHANGE	String	<p>Identifies beginning of new message and session protocol version by means of a session profile identifier (see FIX Session Layer for details). ALWAYS FIRST FIELD IN MESSAGE. (Always unencrypted).</p> <p>Valid values: FIX.4.2 = Session profile FIX.4.2 FIX.4.4 = Session profile FIX4 FIXT.1.1 = Session profile FIXT or LFIXT [Elaboration: The choice between FIXT and LFIXT is subject to counterparty agreement.]</p>	@BeginString	
1128	ApplVerID	CHANGE	String	<p>Specifies the service pack release application layer version being applied at the message level. Enumerated field with values assigned at time of service pack release</p>	@ApplVerID	

Appendix B - Glossary Entries

Term	Definition	Field where used

Appendix C - Abbreviations

Term	Proposed Abbreviation	Proposed Messages, Components, Fields where used

Appendix D - Usage Examples

Example 1: New enumeration value

- FIX session runs FIX 4.2 (BeginString(8)="FIX.4.2")
- Business requirement to support SEC rule change (SR-FINRA-2012-026) relating to the handling of stop and stop limit orders
- A stop order that is triggered by a bid or offer price movement (quote) at which point the stopped order becomes a market/limit order, also known as "stop [limit] on quote" in the US
- FIX Latest is to use new values of OrdType(40) that were added by EP166
 - BeginString(8)="FIX.4.2"
 - OrdType(40)=R (Stop on Bid or Offer) a.k.a. "stop on quote" in the US
 - OrdType(40)=S (Stop Limit on Bid or Offer) a.k.a. "stop limit on quote" in the US

Example 2: Repeating group structure

- FIX session runs FIX 4.2 (BeginString(8)="FIX.4.2")
- Application currently uses ExecBroker(76) to identify the executing firm
- MiFID II requirement to convey LEI of the executing firm as well as the investment decision maker
- FIX Latest is to move away from ExecBroker(76) and use the Parties component instead
 - BeginString(8)="FIX.4.2"
 - NoPartyIDs(453)=2
 - PartyRole(452)=1 (Executing firm), PartyIDSource(447)=N (LEI)
 - PartyRole(452)=122 (Investment decision maker)

Example 3: New message type

- FIX session runs FIX 4.2 (BeginString(8)="FIX.4.2")
- Business requirement to support trade reports in addition to execution reports
- Choices
 - Use a user-defined message TradeCaptureReport(35=UAE)
 - Use the standard message TradeCaptureReport(35=AE) with its required fields and values
- FIX Latest is to use the TradeCaptureReport(35=AE) message as-is
 - BeginString(8)="FIX.4.2" in both ExecutionReport(35=8) and TradeCaptureReport(35=AE)

Appendix E – Disposition of Public Comments

Public comments from Christoph John on November 26, 2021

1. According to <https://www.fixtrading.org/standards/fix-session-layer-online#fix-session-profiles> for FIX4 session profile it is assumed that FIX Latest is used when tag DefaultApplVerID(1137) is absent. Is there anything that differentiates a “normal” FIX.4.4 session from a session using the FIX4 session profile?

Response GTC: No change. FIX4 session profile is the “normal” FIX 4.4 session. The GTC is not trying to create something different here and rather wants to reflect the fact that a typical FIX 4.4 interface will be using tags from higher versions.

2. Because DefaultApplVerID will not be used for FIX.4.2 you cannot make any assumptions (e.g. when looking at a log file) about if FIX Latest is used on a FIX.4.2 connection or not? This is simply agreed between counterparties, correct?

Response GTC: No change. Yes, the GTC recommends that this “agreement” be made in a machine-readable format that precisely defines what is supported. A PDF document should be generated from the machine-readable format. Today, somebody writes a document which then gets keyed in by software developers and testers. That is error-prone and a significant effort compared to the usage of Orchestra and Tablature.

3. So basically speaking this Gap Analysis proposes what counterparties are probably doing anyway: to take fields from newer FIX versions (speak FIX Latest) and use them with FIX4.2 or FIX4.4?

Response GTC: No change. Yes, the GTC wants to provide the formal basis for what is already out there. BeginString(8) having more than one value supports this.

Public comments from Phillip Whitehouse on November 29, 2021

I’m definitely in two minds about this.

On a practical level, what we see when we inter-op with various institutions it is definitely the case that this is what they do. So in terms of making documentation match reality, this change definitely does that.

Our internal FIX engine is forced as a consequence to make no assumption about FIX session capabilities based on BeginString - if you pick a “dictionary” based validation method then we start from the FIX.4.2 spec or the FIX.4.4 spec but we also allow a custom dictionary which written either from customer specification or more commonly, adapted from the FIX.4.x spec version during a time-intensive conformance phase. The BeginString becomes little more than a session identifier like “TargetCompID”.

It would be nice to see Orchestra based conformance but right now we’re not seeing demand for it (but in early phases of adding support in hopeful anticipation).

Even for FIXT, we broadly only see a single application version running on top, so it’s much the same, only with changes made to application “dictionary” and not the transport.

On the other hand, this is sort of anti-specing - and eroding the specifications which is disappointing.

Response GTC: No change.

One cannot assume a specific FIX version when BeginString(8) = “FIXT.1.1”. It could be any of FIX 5.0, FIX 5.0 SP1, FIX 5.0 SP2 or FIX Latest. Only legacy versions below FIX 5.0 have a value in BeginString that matches 1:1 with a FIX version number. FIX was monolithic in nature prior to FIX 5.0 introducing Transport Independence.

One can continue to make assumptions about FIX session capabilities based on BeginString(8). As of FIX 5.0 one can no longer make assumptions on FIX application capabilities based on BeginString(8) other than that they are FIX 5.0 or higher. FIX Latest is not about the session layer, even if the latest session messages still show up in FIXimate for convenience.

The point about supported capabilities is valid, i.e. the fact that the FIX Session Layer specification does not have explicit appendices for the messages, fields and code sets (aka values) for the different session

profiles. It is worth a discussion whether, for example, the capability of the Logon(35=A) having a username and password (added after FIX 4.2) should be allowed when BeginString(8)="FIX.4.2". How do you handle that today?

Response Phillip Whitehouse:

For the FIXT deployments we'd be doing the same but based on the ApplVerID(1128). So then we'd use the relevant spec-generated dictionary as a starting point and then blend in functionality from later EPs / custom tags when needed.

Essentially the BeginString (FIX4) / ApplVerID (FIXT) is a strong hint rather than a definition.

I'm mostly talking about application-layer not session-layer properties. It is the application-layer capabilities that counter-parties tend to want to add to, preserving their original (well tested) session-layer implementation. So that's the area where we are flexible. We use the FIX standards as best practice and guidance - if a counter-party wants a new feature that's in a later version / EP we'll try to adopt the tags and definitions rather than use custom tags for the same purpose.

To be honest we don't have huge use of Username/Password in Logon. In general, and in concert with the FIX community security best practices last time I looked, we tend to authorise connections (which are generally institutional) using TLS client certificates (and/or network-layer security - VPN, private line etc). FIX tags end in logs and audit trails and so forth and obscuring them is ugly. Obviously for some set-ups it may make sense to keep the fields (though TLS certs are getting easier to obtain and handle which might make this less true long term).

However we handle Username/Password checks/insertion beyond the FIX-version specific code, so it would work in FIXT / FIX4.2.

Response GTC: No change.

ApplVerID(1128) is only about the application level. It does not have a value for FIXT, i.e. "ApplVerID (FIXT)" cannot be expressed in the wire format.

Both BeginString(8) and ApplVerID(1128) are only an indication of the capabilities of the session layer and the application layer respectively. That is why the GTC strongly recommends the use of the meta-standard Orchestra to exchange exact definitions by means of an XML file. For legibility, the XML file can be converted to a markdown format.