

FIX Digital Asset WG

Extensions for Digital Asset Trading

May 18, 2022

v0.3

Proposal Status: Public Comment

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.

[](https://creativecommons.org/licenses/by-nd/4.0/)

This work is licensed under a [Creative Commons Attribution-NoDerivatives 4.0 International License](https://creativecommons.org/licenses/by-nd/4.0/).

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

Auto-generate the entire table of contents (press F9) here - or customize up to 3 levels deep.

[Document History 5](#_Toc104051268)

[1 Introduction 6](#_Toc104051269)

[2 Business Requirements 6](#_Toc104051270)

[2.1 Security identifiers for digital assets 6](#_Toc104051271)

[2.1.1 Security Identifier for securities-style trading 7](#_Toc104051272)

[2.1.2 Security identifier for FX-style trading 7](#_Toc104051273)

[2.2 Clarifying Currency for digital asset trading 8](#_Toc104051274)

[2.2.1 Currency field definition 8](#_Toc104051275)

[2.2.2 Currency datatype definition 9](#_Toc104051276)

[2.2.3 Identifying Currency source scheme 9](#_Toc104051277)

[2.3 Extending security types for digital assets 14](#_Toc104051278)

[2.4 Wallet identifier 15](#_Toc104051279)

[3 Issues and Discussion Points 15](#_Toc104051280)

[3.1 Pre- and post-trade 15](#_Toc104051281)

[3.2 Multileg instruments 15](#_Toc104051282)

[4 Proposed Message Flow 15](#_Toc104051283)

[5 FIX Message Tables 16](#_Toc104051284)

[6 FIX Component Blocks 16](#_Toc104051285)

[6.1 Component SecAltIDGrp 20](#_Toc104051286)

[6.2 Component LegSecAltIDGrp 21](#_Toc104051287)

[6.3 Component UndSecAltIDGrp 22](#_Toc104051288)

[7 Category Changes 22](#_Toc104051289)

[Appendix A - Data Dictionary 23](#_Toc104051290)

[Appendix B - Glossary Entries 42](#_Toc104051291)

[Appendix C - Abbreviations 42](#_Toc104051292)

[Appendix D - Usage Examples 43](#_Toc104051293)

Table of Tables

[Table 1 Fields to identify currency source schemes in messages 10](#_Toc103068590)

[Table 2 Fields to identify currency source schemes in components 11](#_Toc103068591)

A Table of Figures is not required. If used, use styles to tag the captions and auto-generate the list here. If not used, remove this section.

# Document History

| **Revision** | **Date** | **Author** | **Revision Comments** |
| --- | --- | --- | --- |
| v0.1 | Apr 26, 2022 | Hanno Klein, GTC  Shraddha Gohad, IIT | Initial draft |
| v0.2 | May 11, 2022 | Hanno Klein, GTC | Updates after WG review Apr 28  Updates of Chapters 5 and 6 |
| V0.3 | May 12, 2022  May 18, 2022  May 20, 2022 | Hanno Klein, GTC | Updates after WG review May 12  Updated Data Dictionary based on the requirements.  Added examples from Digital Asset WG. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

The above document history section, including date, author, and comments, is required to track editing changes to the document. List revisions in **ascending order**. Please insert additional rows in the table as needed.

Template version information:

r2: 2010-11-13 Revised to support abbreviations, inlined component references

r3: 2011-12-02 Revised to add additional usage clarification

r3.1: 2013-01-04 Revised Copyright year, changed template to Office 2013 .docx version, updated font to Calibri for cleaner look. Added additional template usage clarification.

R3.2 2016-05-23 – revised the copyright date, corrected document references and replaced the FIX logo with the most current graphic.

2016-09-14 – Removed blue highlighting from hidden text content, removed ICR column from 5.1 and 6.1 tables and added “FIX Spec” to the Comments column header of table 6.1 and revised notation of optional elaboration for fields and enums in the data dictionary.

# Introduction

Provide an introduction to the content, purpose, or impetus of the proposal; the business need / problem being solved; and the scope. Include and label any references, supporting documentation, and related proposals. If the proposal is based on existing implementations, describe them here in the appropriate subsections. It is recommended that a "Summary of Proposed Changes" sub-section be provided within this section.

FIX provides support for a large number of asset classes. The FIX Digital Asset Working Group was created to provide standardization for digital assets, not only including cryptocurrencies like Bitcoin but also any kind of digital tokens that can be traded between buy-side and sell-side or on electronic venues.

The FIX Digital Asset Working Group further wants to facilitate industry-wide implementation of standardized trading of digital assets by providing recommended practices. This work is going on in parallel to this gap analysis, which seeks to propose extensions to FIX to cover new business requirements arising from the nature of digital assets trading. The primary focus of this proposal is on the trading area with the objective to address the areas of pre- and post-trade with future proposals.

The proposal suggests FIX to support two different trading styles due to different existing market practices for digital asset trading.

* FX-style – currency pairs (e.g. BTC/USD, BTC/ETH) of fiat and/or digital currencies
* Securities-style – for trading any digital asset, priced and/or settling in either a fiat or digital currency

The identification of fiat currencies has been standardized a long time ago with ISO 4217, whereas the identification of digital assets (including digital currencies) has only recently been standardized with ISO 24165 *Digital Token Identifier*. One of the main proposals in this gap analysis is to extend the FIX datatype "Currency" to support both ISO 4217 and ISO 24165. This requires a number of additional extensions to support the ability to distinguish between the two code schemes when using FIX currency fields such as Currency(15) and SettlCurrency(120).

Furthermore, this gap analysis addresses gaps in FIX related to the taxonomy for digital assets and the identification of wallets.

# Business Requirements

Describe the business problem, requirements or workflow that is the source for the proposed changes to the FIX Protocol. The descriptions should help the Global Technical Committee understand the business areas, business requirements that resulted in the proposal and context for the proposal. The descriptions should use business or layman's terms. Business flow diagrams may also be helpful.

## Security identifiers for digital assets

Digital assets are traded either like a currency exchange or a security. FIX is able to support both approaches of how market participants trades digital assets.

In traditional currency exchange or Foreign Exchange trading the Symbol(55) field carries the currency pair using the "CCY1/CCY2" convention, e.g. EUR/USD. Many digital asset market participants who have adopted FIX have been using this FX-style convention, e.g. "BTC/USD"

An alternative convention adopted by some digital asset market participants is trading a digital asset like a security similar to trading of stocks. This style of trading simply provides the digital asset "symbol" in the Symbol(55) field, e.g. "BTC". This approach is popular when the digital asset is priced in terms of a fiat (government issued or sovereign) currency, such as US Dollars.

Note that how a digital asset trades have no bearing on the function or classification of a digital asset. Digital assets that are considered securities can trade using Foreign Exchange convention, and digital assets that are considered virtual currencies can trade like stocks. FIX makes no recommendation regarding which of the two approaches should be used; this is left to bilateral agreement.

The business requirements are to support the identification and trading of digital assets under either one of these styles. Recently, existing symbologies (e.g. ISIN and FIGI) have or soon will support digital asset identification, and new symbologies (e.g. ISO 24165 Digital Token Identifier (DTI)) exclusively support digital assets. They provide more precise and unique identification of the digital asset and reduce the need to depend upon exchange symbology, which frequently is not consistent across exchanges.

The sections below go into details based on the trading style and proposed solution.

### Security Identifier for securities-style trading

Securities-style trading is already supported by FIX with the exception of the ability to identify a value in SecurityID(48) as a DTI.

SecurityIDSource(22) conveys the identifier scheme of the value in SecurityID(48). It is proposed to add a new value for ISO 24165 (Digital Token Identifier – DTI). The values of SecurityIDSource(22) are identified in the Orchestra XML file with SecurityIDSourceCodeSet that is also used by SecurityAltIDSource(456) and many other fields. The extension of SecurityIDSourceCodeSet automatically extends all fields using this code set.

TBD = Digital Token Identifier (DTI) – ISO 24165

### Security identifier for FX-style trading

Currency symbols used for FX-style trading use ISO 4217 3-character codes which are created only under the framework of the standard. Digital assets did not have the benefit of an ISO standard at their inception, which has led to different symbols or labels to identify the same digital asset, e.g. "BTC" as well as "XBT" for bitcoin. ISO 24165 (Digital Token Identifier – DTI) was recently published to address the ambiguity by providing a unique identifier for cryptocurrencies as well as other digital assets where one or more random but human-readable labels can be attached. For example, the DTI assigned to bitcoin is 4H95J0R2X.

In order to disambiguate Symbol(55) for FX-style trading of digital assets, the requirement is to provide an ability to specify the DTI associated with the non-unique digital asset symbol(s) used in Symbol(55). To meet this requirement, it is proposed that the current concept for SecAltIDGrp be extended, which is currently limited to specifying alternate security identifiers to SecurityID(48) and SecurityIDSource(22), to allow for specifying alternative identifiers associated with the symbology in Symbol(55).

The proposed solution would support an explicit DTI or other security identifier that references either the first or the second symbol specified in the Symbol(55) field in FX-style trading via an ordinal reference within the SecAltIDGrp. This avoids the need for the definition of an implicit ordering of repeating group instances for SecAltIDGrp. It is proposed to add a new field to SecAltIDGrp for an ordinal reference as follows.

SymbolPositionNumber(TBD) Reference to the first or second currency or digital asset in Symbol(55) for FX-style trading. Conditionally required when one or both symbols in Symbol(55) represent a digital asset.

The values are proposed as 1 = First currency or digital asset in Symbol(55) and 2 = Second currency or digital asset in Symbol(55).

SecurityAltID(455) and SecurityAltIDSource(456) are both required when using SecAltIDGrp.

SecurityAltIDSource(456) has the same values as SecurityIDSource(22) and is used to denote the fiat currency with 6 = ISO Currency Code or another symbology that supports digital assets, such as TBD=Digital Token Identifier (see section 2.1.1). The following table provides some examples. Note that Symbol(55) uses exchange ticker or other bilaterally agreed symbology, which is not consistent. In the example below, Bitcoin could be represented as either BTC or XBT in the Symbol(55) field, but the DTI for Bitcoin is the same in all cases.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Buying 2.5 Bitcoin in USD** | **Buying $100,000 of Bitcoin (or selling $100,000 for Bitcoin)** | **Buying 10 Ether in Bitcoin** |
| Symbol(55) | BTC/USD | BTC/USD | ETH/XBT |
| NoSecurityAltID(454) | 2 | 2 | 2 |
| > SecurityAltID(455) | 4H95J0R2X  (Bitcoin’s DTI) | 4H95J0R2X  (Bitcoin’s DTI) | X9J9K8725  (Ether’s DTI) |
| > SecurityAltIDSource(456) | TBD = DTI | TBD = DTI | TBD = DTI |
| > SymbolPositionNumber(TBD) | 1 | 1 | 1 |
| > SecurityAltID(455) | USD | USD | 4H95J0R2X  (Bitcoin’s DTI) |
| > SecurityAltIDSource(456) | 6 = ISO Currency | 6 = ISO Currency | TBD = DTI |
| > SymbolPositionNumber(TBD) | 2 | 2 | 2 |

## Clarifying Currency for digital asset trading

### Currency field definition

The current description of the FIX field Currency(15) is as follows:

*Identifies currency used for price. Absence of this field is interpreted as the default for the security. It is recommended that systems provide the currency value whenever possible. See Appendix 6-A: Valid Currency Codes for information on obtaining valid values.*

This description has a securities trading focus due to the historical beginnings of FIX for equities trading. It is proposed to update this definition to clarify how this field applies to FX (fiat currencies and digital assets) and the securities type of trading digital assets. The FIX community of FX users have understood that the use of Currency(15) denotes the dealing currency of quantity fields such as OrderQty(38) (i.e. the value in the quantity field is denominated in that specified currency code) and Price(44) represents the conversion rate between the two currencies. The new definition should include text to explain the relationship with the new CurrencyCodeSource(TBD) field (see 2.2.3 *Identifying Currency source scheme*) and add specific information about digital asset trading.

Orchestra allows a high-level definition (a.k.a synopsis or description) as well an elaboration to provide additional detail. It is proposed to use that capability and change the text as follows.

**Synopsis/Description** – Identifies currency used for price or quantity fields, depending on the asset class being traded. CurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent.

**Elaboration** – For securities trading and digital assets traded securities-style, identifies the currency used to denote the price. Absence of this field is interpreted as the default for the security. For Foreign Exchange (FX) and digital assets traded FX-style, identifies the dealt currency used to denominate (the) quantity related field(s).

### Currency datatype definition

The FIX datatype "Currency" is used for a large number of fields and historically used to convey fiat currencies (ISO 4217). It currently is defined as follows:

*String field representing a currency type using ISO 4217 Currency code (3 character).*

This historical definition precludes the use of the FIX currency fields for digital asset trading. It is proposed to extend the definition of the datatype as follows.

*String field representing a currency type using ISO 4217 CurrencyCode (3 character) or a digital asset. The digital asset may be denoted in one of several schemes, including but not limited to ISO 24165 Digital Token Identifier. The default is ISO 4217 in the absence of a second field identifying the source scheme.*

### Identifying Currency source scheme

The extension of the scope of the FIX datatype "Currency" (see section 2.2) introduces a possible ambiguity related to the domain of values in the FIX currency fields. Whilst the absence of any further information in a message should indicate the use of ISO 4217 as a default, an optional explicit approach is needed for digital asset trading.

It is proposed to add new fields for those FIX currency related fields that are relevant for the entire life-cycle of digital assets to be able to explicitly define the source scheme used. See section 3 for further elaborations on scope of this gap analysis.

The following table proposes new fields for currency source schemes and the messages they are to be added to. It shows the existing currency related field that is already part of the listed messages followed by the proposed new field.

Table 1 Fields to identify currency source schemes in messages

| **Field** | **Add to message(s)** |
| --- | --- |
| Currency(15)  CurrencyCodeSource(TBD) | * Advertisement(35=7) * IOI(35=6) * BidRequest(35=k) * Quote(35=S) * QuoteResponse(35=AJ) * QuoteStatusReport(35=AI) * SecurityDefinitionRequest(35=c) * SecurityDefinition(35=d) * SecurityDefinitionUpdateReport(35=BP) * SecurityListRequest(35=x) * DerivativeSecurityListRequest(35=z) * SecurityStatusRequest(35=e) * SecurityStatus(35=f) * MarketDefinition(35=BU) * MarketDefinitionUpdateReport(35=BV) * MarketDataStatisticsReport(35=DP) * PartyRiskLimitCheckRequest(35=DF) * PartyRiskLimitCheckRequestAck(35=DG) * NewOrderSingle(35=D) * NewOrderMultileg(35=AB) * OrderCancelReplaceRequest(35=G) * MultilegOrderCancelReplace(35=AC) * ExecutionReport(35=8) * NewOrderCross(35=s) * CrossOrderCancelReplaceRequest(35=t) * TradeCaptureReport(35=AE) * TradeCaptureReportAck(35=AR) * TradeAggregationRequest(35=DW) * RequestForPositions(35=AN) * RequestForPositionsAck(35=AO) * PositionMaintenanceRequest(35=AL) * PositionMaintenanceReport(35=AM) * PositionReport(35=AP) * PositionTransferInstruction(35=DL) * PositionTransferReport(35=DN) * AllocationInstruction(35=J) * AllocationInstructionAlert(35=BM) * AllocationReport(35=AS) * Confirmation(35=AK) * AssignmentReport(35=AW) * CollateralRequest(35=AX) * CollateralAssignment(35=AY) * CollateralResponse(35=AZ) * CollateralReport(35=BA) * CollateralInquiry(35=BB) * CollateralInquiryAck(35=BG) * MarginRequirementReport(35=CJ) |
| SettlCurrency(120)  SettlCurrencyCodeSource(TBD) | * Quote(35=S) * NewOrderSingle(35=D) * NewOrderMultileg(35=AB) * OrderCancelReplaceRequest(35=G) * MultilegOrderCancelReplace(35=AC) * ExecutionReport(35=8) * TradeCaptureReport(35=AE) * TradeCaptureReportAck(35=AR) * RequestForPositions(35=AN) * RequestForPositionsAck(35=AO) * PositionMaintenanceRequest(35=AL) * PositionMaintenanceReport(35=AM) * PositionReport(35=AP) * Confirmation(35=AK) * SettlementInstructionRequest(35=AV) |
| SettlPriceUnitOfMeasureCurrency(1887)  SettlPriceUnitOfMeasureCurrencyCodeSource(TBD) | * PositionReport(35=AP) |

The following table proposes new fields for currency source schemes and the components they are to be added to. It shows the existing currency related field that is already part of the listed components followed by the proposed new field.

Table 2 Fields to identify currency source schemes in components

|  |  |
| --- | --- |
| **Field** | **Add to component(s)** |
| Currency(15)  CurrencyCodeSource(TBD) | * InstrmtMDReqGrp * InstrmtMatchSideGrp * InstrmtStrkPxGrp * ListOrdGrp * MDFullGrp * MDIncGrp * QuotEntryAckGrp * QuotEntryGrp * QuotReqGrp * QuotReqRjctGrp * RelSymDerivSecGrp * RelSymDerivSecUpdGrp * SecListGrp * SecLstUpdRelSymGrp * SettlObligationInstructions |
| LegCurrency(556)  LegCurrencyCodeSource(TBD) | * InstrumentLeg |
| SettlCurrency(120)  SettlCurrencyCodeSource(TBD) | * AllocGrp * InstrmtMatchSideGrp * ListOrdGrp * MDFullGrp * MDIncGrp * QuotReqGrp * SettlInstGrp * SettlObligationInstructions * SideCrossOrdModGrp |
| LegSettlCurrency(675)  LegSettlCurrencyCodeSource(TBD) | * InstrmtLegExecGrp * LegOrdGrp * SideCrossLegGrp * TrdInstrmtLegExecGrp * TrdInstrmtLegGrp |
| SideCurrency(1154)  SideCurrencyCodeSource(TBD) | * TrdCapRptSideGrp * TrdCapRptAckSideGrp * TrdMatchSideGrp |
| SideSettlCurrency(1155)  SideSettlCurrencyCodeSource(TBD) | * TrdCapRptSideGrp * TrdCapRptAckSideGrp * TrdMatchSideGrp |
| SettlementAmountCurrency(1702)  SettlementAmountCurrencyCodeSource(TBD) | * SettlementAmountGrp |
| StrikeCurrency(947)  StrikeCurrencyCodeSource(TBD) | * Instrument |
| UnitOfMeasureCurrency(1716)  UnitOfMeasureCurrencyCodeSource(TBD) | * Instrument |
| PriceUnitOfMeasureCurrency(1717)  PriceUnitOfMeasureCurrencyCodeSource(TBD) | * Instrument |
| PriceQuoteCurrency(1524) PriceQuoteCurrencyCodeSource(TBD) | * Instrument |
| LegStrikeCurrency(942)  LegStrikeCurrencyCodeSource(TBD) | * InstrumentLeg |
| LegUnitOfMeasureCurrency(1720)  LegUnitOfMeasureCurrencyCodeSource(TBD) | * InstrumentLeg |
| LegPriceUnitOfMeasureCurrency(1721)  LegPriceUnitOfMeasureCurrencyCodeSource(TBD) | * InstrumentLeg |
| LegPriceQuoteCurrency(1528) LegPriceQuoteCurrencyCodeSource(TBD) | * InstrumentLeg |
| DerivativeStrikeCurrency(1262)  DerivativeStrikeCurrencyCodeSource(TBD) | * DerivativeInstrument |
| DerivativeUnitOfMeasureCurrency(1722)  DerivativeUnitOfMeasureCurrencyCodeSource(TBD) | * DerivativeInstrument |
| DerivativePriceUnitOfMeasureCurrency(1723)  DerivativePriceUnitOfMeasureCurrencyCodeSource(TBD) | * DerivativeInstrument |
| DerivativePriceQuoteCurrency(1576) DerivativePriceQuoteCurrencyCodeSource(TBD) | * DerivativeInstrument |
| UnderlyingCurrency(318)  UnderlyingCurrencyCodeSource(TBD) | * UnderlyingInstrument |
| UnderlyingStrikeCurrency(941)  UnderlyingStrikeCurrencyCodeSource(TBD) | * UnderlyingInstrument |
| UnderlyingUnitOfMeasureCurrency(1718)  UnderlyingUnitOfMeasureCurrencyCodeSource(TBD) | * UnderlyingInstrument |
| UnderlyingPriceUnitOfMeasureCurrency(1719)  UnderlyingPriceUnitOfMeasureCurrencyCodeSource(TBD) | * UnderlyingInstrument |
| UnderlyingPriceQuoteCurrency(1526) UnderlyingPriceQuoteCurrencyCodeSource(TBD) | * UnderlyingInstrument |
| UnderlyingNotionalCurrency(2615)  UnderlyingNotionalCurrencyCodeSource(TBD) | * UnderlyingInstrument |
| CommCurrency(479)  CommCurrencyCodeSource(TBD) | * CommissionData |
| CommissionCurrency(2643)  CommissionCurrencyCodeSource(TBD) | * CommissionDataGrp |
| CommissionUnitOfMeasureCurrency(2645)  CommissionUnitOfMeasureCurrencyCodeSource(TBD) | * CommissionDataGrp |
| AllocCommissionCurrency(2657)  AllocCommissionCurrencyCodeSource(TBD) | * AllocCommissionDataGrp |
| AllocCommissionUnitOfMeasureCurrency(2659)  AllocCommissionUnitOfMeasureCurrencyCodeSource(TBD) | * AllocCommissionDataGrp |
| AllocSettlCurrency(736)  AllocSettlCurrencyCodeSource(TBD) | * PreAllocGrp * PreAllocMlegGrp * AllocGrp * TrdAllocGrp |
| LegAllocSettlCurrency(1367)  LegAllocSettlCurrencyCodeSource(TBD) | * LegPreAllocGrp |
| CollateralCurrency(1705)  CollateralCurrencyCodeSource(TBD) | * CollateralAmountGrp |
| SideCollateralCurrency(2695)  SideCollateralCurrencyCodeSource(TBD) | * SideCollateralAmountGrp |
| CollateralReinvestmentCurrency(2843)  CollateralReinvestmentCurrencyCodeSource(TBD) | * CollateralReinvestmentGrp |
| SideCollateralReinvestmentCurrency(2866)  SideCollateralReinvestmentCurrencyCodeSource(TBD) | * SideCollateralReinvestmentGrp |
| TradeAllocCurrency(1847)  TradeAllocCurrencyCodeSource(TBD) | * TradeAllocAmtGrp |
| TradingCurrency(1245)  TradingCurrencyCodeSource(TBD) | * BaseTradingRules |
| LimitAmtCurrency(1634)  LimitAmtCurrencyCodeSource(TBD) | * LimitAmts |
| PosQtyUnitOfMeasureCurrency(1835)  PosQtyUnitOfMeasureCurrencyCodeSource(TBD) | * PositionQty |
| PositionCurrency(1055)  PositionCurrencyCodeSource(TBD) | * PositionAmountData |
| LegPosCurrency(1589)  LegPosCurrencyCodeSource(TBD) | * LegPositionAmountData |
| RiskLimitCurrency(1532)  RiskLimitCurrencyCodeSource(TBD) | * RiskLimitTypesGrp |
| EntitlementAttribCurrency(1781)  EntitlementAttribCurrencyCodeSource(TBD) | * EntitlementAttribGrp |
| ComplexOptPayoutCurrency(2122)  ComplexOptPayoutCurrencyCodeSource | * ComplexEvents |
| ComplexEventCurrencyOne(2124)  ComplexEventCurrencyOneCodeSource(TBD) | * ComplexEvents |
| ComplexEventCurrencyTwo(2125)  ComplexEventCurrencyTwoCodeSource(TBD) | * ComplexEvents |
| LegComplexOptPayoutCurrency(2226)  LegComplexOptPayoutCurrencyCodeSource | * LegComplexEvents |
| LegComplexEventCurrencyOne(2233)  LegComplexEventCurrencyOneCodeSource(TBD) | * LegComplexEvents |
| LegComplexEventCurrencyTwo(2234)  LegComplexEventCurrencyTwoCodeSource(TBD) | * LegComplexEvents |
| UnderlyingComplexOptPayoutCurrency(2266)  UnderlyingComplexOptPayoutCurrencyCodeSource(TBD) | * UnderlyingComplexEvents |
| UnderlyingComplexEventCurrencyOne(2268)  UnderlyingComplexEventCurrencyOneCodeSource(TBD) | * UnderlyingComplexEvents |
| UnderlyingComplexEventCurrencyTwo(2269)  UnderlyingComplexEventCurrencyTwoCodeSource(TBD) | * UnderlyingComplexEvents |
| BenchmarkCurveCurrency(220)  BenchmarkCurveCurrencyCodeSource(TBD) | * SpreadOrBenchmarkCurveData |
| LegBenchmarkCurveCurrency(676)  LegBenchmarkCurveCurrencyCodeSource(TBD) | * LegBenchmarkCurveData |
| AgreementCurrency(918)  AgreementCurrencyCodeSource(TBD) | * FinancingDetails |
| LegAgreementCurrency(2495)  LegAgreementCurrencyCodeSource(TBD) | * LegFinancingDetails |
| FundingSourceCurrency(2847)  FundingSourceCurrency CodeSource(TBD) | * FundingSourceGrp |
| PayCollectCurrency(1709)  PayCollectCurrencyCodeSource(TBD) | * PayCollectGrp |
| PostTradePaymentCurrency(2818)  PostTradePaymentCurrencyCodeSource(TBD) | * PostTradePayment |

All currency fields for OTC trading are out of scope of this document but may be subject to a future gap analysis.

All of these new fields should use a subset of the values available for SecurityIDSource(22) (defined by SecurityIDSourceCodeSet), depending on their applicability to digital assets. The code set values themselves are proposed to be identical to those used in SecurityIDSource(22), e.g. SecurityIDSource(22) and CurrencyCodeSource(TBD) both use 6 = ISO Currency Code for fiat currencies. The following table proposes the list applicable values for currency source schemes.

|  |  |
| --- | --- |
| **Value** | **Name** |
| 1 | CUSIP |
| 2 | SEDOL |
| 4 | ISIN |
| 6 | ISO Currency Code |
| S | Financial Instrument Global Identifier (FIGI) |
| TBD | Digital Token Identifier (DTI) |

## Extending security types for digital assets

FIX provides fields such as Product(460), CFICode(461), SecurityType(167), SecuritySubType(762) in the Instrument component to identify products and instruments, i.e. describing "what" is being traded. It is proposed to add a new security type for digital assets trading by adding the following value to SecurityType(167) (and related fields using SecurityTypeCodeSet).

DIGITAL = Digital Asset

*Elaboration*: Asset that exists only in digital form or which is the digital representation of another asset (Source: ISO 24165 - Terms and Definitions).

## Wallet identifier

Digital asset trading typically uses a "wallet" to store the digital assets belonging to the beneficiary or owner. The "wallet" owner can be identified with the Parties component using an appropriate party role. The format of this identifier is specific to the type of digital asset. It is proposed to convey the wallet identifier in PtysSubGrp, the sub-component of Parties, so that information can be attached to the appropriate party role and its identifier. To support this, a new value to PartySubIDType(803) (and related fields using PartySubIDTypeCodeSet) is proposed as follows:

TBD = Wallet identifier

# Issues and Discussion Points

The information in this section can be presented in table or numbered list format or sub-sections of descriptive text. Include issues and important discussion points that arose during the sub-committee or working group's effort to develop the gap analysis proposal. Also include resolutions of the issues and discussion points. The items will aid in understanding the thought process and tracks for the decisions made.

## Pre- and post-trade

This gap analysis is focused on digital asset trading.  Pre-trade and post-trade workflows for digital assets have not been discussed or considered as part of this gap analysis.

The Working Group decided to simply provide the new fields to express currency code schemes for all relevant fields including fields used in pre-trade and post-trade workflows. Discussions specific to pre-trade and post-trade workflows will be conducted in the future.  Fields used for OTC (e.g. OTC swaps) are also excluded in this gap analysis.

## Multileg instruments

The gap analysis has a focus on single leg instruments and does not include messages or components for the trading of multileg instruments. The issue is whether any of these should be included in this initial gap analysis or deferred to a future gap analysis related to digital asset trading.

The working group decided to include leg level currency fields into the scope of new fields to express currency code schemes. There is no reason to believe that digital assets will not be traded in conjunction with strategies.

# Proposed Message Flow

NONE

# FIX Message Tables

This gap analysis impacts a large number of messages as it adds fields to identify currency source schemes to their root level. However, the new fields are always positioned right after the existing currency field. Both the existing and the new fields and the corresponding messages are listed in *Table 1 Fields to identify currency source schemes in messages*.

# FIX Component Blocks

This gap analysis impacts a large number of components as it adds fields to identify currency source schemes. However, the new fields are always positioned right after the existing currency field. Both the existing and the new fields and the corresponding components are listed in *Table 2 Fields to identify currency source schemes in components*. The following table shows the components and the existing and new field(s) for each of them.

| **Component** | **Currency field(s)** |
| --- | --- |
| Instrument | StrikeCurrency(947)  StrikeCurrencyCodeSource(TBD)  UnitOfMeasureCurrency(1716)  UnitOfMeasureCurrencyCodeSource(TBD)  PriceUnitOfMeasureCurrency(1717)  PriceUnitOfMeasureCurrencyCodeSource(TBD)  PriceQuoteCurrency(1524)  PriceQuoteCurrencyCodeSource(TBD) |
| InstrumentLeg | LegCurrency(556)  LegCurrencyCodeSource(TBD)  LegStrikeCurrency(942)  LegStrikeCurrencyCodeSource(TBD)  LegUnitOfMeasureCurrency(1720)  LegUnitOfMeasureCurrencyCodeSource(TBD)  LegPriceUnitOfMeasureCurrency(1721)  LegPriceUnitOfMeasureCurrencyCodeSource(TBD)  LegPriceQuoteCurrency(1528)  LegPriceQuoteCurrencyCodeSource(TBD) |
| UnderlyingInstrument | UnderlyingCurrency(318)  UnderlyingCurrencyCodeSource(TBD)  UnderlyingStrikeCurrency(941)  UnderlyingStrikeCurrencyCodeSource(TBD)  UnderlyingUnitOfMeasureCurrency(1718)  UnderlyingUnitOfMeasureCurrencyCodeSource(TBD)  UnderlyingPriceUnitOfMeasureCurrency(1719)  UnderlyingPriceUnitOfMeasureCurrencyCodeSource(TBD)  UnderlyingPriceQuoteCurrency(1526)  UnderlyingPriceQuoteCurrencyCodeSource(TBD)  UnderlyingNotionalCurrency(2615)  UnderlyingNotionalCurrencyCodeSource(TBD) |
| DerivativeInstrument | DerivativeStrikeCurrency(1262)  DerivativeStrikeCurrencyCodeSource(TBD)  DerivativeUnitOfMeasureCurrency(1722)  DerivativeUnitOfMeasureCurrencyCodeSource(TBD)  DerivativePriceUnitOfMeasureCurrency(1723)  DerivativePriceUnitOfMeasureCurrencyCodeSource(TBD)  DerivativePriceQuoteCurrency(1576)  DerivativePriceQuoteCurrencyCodeSource(TBD) |
| CommissionDataGrp | CommissionCurrency(2643)  CommissionCurrencyCodeSource(TBD)  CommissionUnitOfMeasureCurrency(2645)  CommissionUnitOfMeasureCurrencyCodeSource(TBD) |
| CommissionData | CommCurrency(479)  CommCurrencyCodeSource(TBD) |
| AllocCommissionDataGrp | AllocCommissionCurrency(2657)  AllocCommissionCurrencyCodeSource(TBD)  AllocCommissionUnitOfMeasureCurrency(2659)  AllocCommissionUnitOfMeasureCurrencyCodeSource(TBD) |
| PreAllocGrp | AllocSettlCurrency(736)  AllocSettlCurrencyCodeSource(TBD) |
| PreAllocMlegGrp | AllocSettlCurrency(736)  AllocSettlCurrencyCodeSource(TBD) |
| AllocGrp | AllocSettlCurrency(736)  AllocSettlCurrencyCodeSource(TBD)  SettlCurrency(120)  SettlCurrencyCodeSource(TBD) |
| TrdAllocGrp | AllocSettlCurrency(736)  AllocSettlCurrencyCodeSource(TBD) |
| LegPreAllocGrp | LegAllocSettlCurrency(1367)  LegAllocSettlCurrencyCodeSource(TBD) |
| CollateralAmountGrp | CollateralCurrency(1705)  CollateralCurrencyCodeSource(TBD) |
| SideCollateralAmountGrp | SideCollateralCurrency(2695)  SideCollateralCurrencyCodeSource(TBD) |
| CollateralReinvestmentGrp | CollateralReinvestmentCurrency(2843)  CollateralReinvestmentCurrencyCodeSource(TBD) |
| SideCollateralReinvestmentGrp | SideCollateralReinvestmentCurrency(2866)  SideCollateralReinvestmentCurrencyCodeSource(TBD) |
| TradeAllocAmtGrp | TradeAllocCurrency(1847)  TradeAllocCurrencyCodeSource(TBD) |
| BaseTradingRules | TradingCurrency(1245)  TradingCurrencyCodeSource(TBD) |
| LimitAmts | LimitAmtCurrency(1634)  LimitAmtCurrencyCodeSource(TBD) |
| PositionQty | PosQtyUnitOfMeasureCurrency(1835)  PosQtyUnitOfMeasureCurrencyCodeSource(TBD) |
| PositionAmountData | PositionCurrency(1055)  PositionCurrencyCodeSource(TBD) |
| LegPositionAmountData | LegPosCurrency(1589)  LegPosCurrencyCodeSource(TBD) |
| RiskLimitTypesGrp | RiskLimitCurrency(1532)  RiskLimitCurrencyCodeSource(TBD) |
| EntitlementAttribGrp | EntitlementAttribCurrency(1781)  EntitlementAttribCurrencyCodeSource(TBD) |
| ComplexEvents | ComplexOptPayoutCurrency(2122)  ComplexOptPayoutCurrencyCodeSource  ComplexEventCurrencyOne(2124)  ComplexEventCurrencyOneCodeSource(TBD)  ComplexEventCurrencyTwo(2125)  ComplexEventCurrencyTwoCodeSource(TBD) |
| LegComplexEvents | LegComplexOptPayoutCurrency(2226)  LegComplexOptPayoutCurrencyCodeSource  LegComplexEventCurrencyOne(2233)  LegComplexEventCurrencyOneCodeSource(TBD)  LegComplexEventCurrencyTwo(2234)  LegComplexEventCurrencyTwoCodeSource(TBD) |
| UnderlyingComplexEvents | UnderlyingComplexOptPayoutCurrency(2266)  UnderlyingComplexOptPayoutCurrencyCodeSource  UnderlyingComplexEventCurrencyOne(2268)  UnderlyingComplexEventCurrencyOneCodeSource(TBD)  UnderlyingComplexEventCurrencyTwo(2269)  UnderlyingComplexEventCurrencyTwoCodeSource(TBD) |
| SpreadOrBenchmarkCurveData | BenchmarkCurveCurrency(220)  BenchmarkCurveCurrencyCodeSource(TBD) |
| LegBenchmarkCurveData | LegBenchmarkCurveCurrency(676)  LegBenchmarkCurveCurrencyCodeSource(TBD) |
| FinancingDetails | AgreementCurrency(918)  AgreementCurrencyCodeSource(TBD) |
| LegFinancingDetails | LegAgreementCurrency(2495)  LegAgreementCurrencyCodeSource(TBD) |
| FundingSourceGrp | FundingSourceCurrency(2847)  FundingSourceCurrency CodeSource(TBD) |
| PayCollectGrp | PayCollectCurrency(1709)  PayCollectCurrencyCodeSource(TBD) |
| PostTradePayment | PostTradePaymentCurrency(2818)  PostTradePaymentCurrencyCodeSource(TBD) |
| TrdCapRptSideGrp | SideSettlCurrency(1155)  SideSettlCurrencyCodeSource(TBD)  SideCurrency(1154)  SideCurrencyCodeSource(TBD) |
| TrdCapRptAckSideGrp | SideSettlCurrency(1155)  SideSettlCurrencyCodeSource(TBD)  SideCurrency(1154)  SideCurrencyCodeSource(TBD) |
| TrdMatchSideGrp | SideSettlCurrency(1155)  SideSettlCurrencyCodeSource(TBD)  SideCurrency(1154)  SideCurrencyCodeSource(TBD) |
| InstrmtLegExecGrp | LegSettlCurrency(675)  LegSettlCurrencyCodeSource(TBD) |
| LegOrdGrp | LegSettlCurrency(675)  LegSettlCurrencyCodeSource(TBD) |
| SideCrossLegGrp | LegSettlCurrency(675)  LegSettlCurrencyCodeSource(TBD) |
| TrdInstrmtLegExecGrp | LegSettlCurrency(675)  LegSettlCurrencyCodeSource(TBD) |
| TrdInstrmtLegGrp | LegSettlCurrency(675)  LegSettlCurrencyCodeSource(TBD) |
| InstrmtMatchSideGrp | Currency(15)  CurrencyCodeSource(TBD)  SettlCurrency(120)  SettlCurrencyCodeSource(TBD) |
| ListOrdGrp | Currency(15)  CurrencyCodeSource(TBD)  SettlCurrency(120)  SettlCurrencyCodeSource(TBD) |
| MDFullGrp | Currency(15)  CurrencyCodeSource(TBD)  SettlCurrency(120)  SettlCurrencyCodeSource(TBD) |
| MDIncGrp | Currency(15)  CurrencyCodeSource(TBD)  SettlCurrency(120)  SettlCurrencyCodeSource(TBD) |
| QuotReqGrp | Currency(15)  CurrencyCodeSource(TBD)  SettlCurrency(120)  SettlCurrencyCodeSource(TBD) |
| SettlInstGrp | SettlCurrency(120)  SettlCurrencyCodeSource(TBD) |
| SettlObligationInstructions | Currency(15)  CurrencyCodeSource(TBD)  SettlCurrency(120)  SettlCurrencyCodeSource(TBD) |
| SideCrossOrdModGrp | SettlCurrency(120)  SettlCurrencyCodeSource(TBD) |
| InstrmtMDReqGrp | Currency(15)  CurrencyCodeSource(TBD) |
| InstrmtStrkPxGrp | Currency(15)  CurrencyCodeSource(TBD) |
| QuotEntryAckGrp | Currency(15)  CurrencyCodeSource(TBD) |
| QuotEntryGrp | Currency(15)  CurrencyCodeSource(TBD) |
| QuotReqRjctGrp | Currency(15)  CurrencyCodeSource(TBD) |
| RelSymDerivSecGrp | Currency(15)  CurrencyCodeSource(TBD) |
| RelSymDerivSecUpdGrp | Currency(15)  CurrencyCodeSource(TBD) |
| SecListGrp | Currency(15)  CurrencyCodeSource(TBD) |
| SecLstUpdRelSymGrp | Currency(15)  CurrencyCodeSource(TBD) |

## Component SecAltIDGrp

The following table is required for newly proposed components only; it is not required for components that are being modified by your proposal unless you are proposing new or updating component synopsis or elaboration. When proposing new or updating component synopsis or elaboration, only those fields should be used.

When proposing a new message type, describe the usage. If any usage is asset type specific, clearly indicate it.

* **Component Name** - The component or repository name (no embedded spaces or punctuation characters).
* **Component Abbreviated Name** (for FIXML) - The abbreviated name of the message using standard FIX abbreviations. If new terms require abbreviation, propose the abbreviation for the new term(s) in Appendix C. Submitter may propose the abbreviations for the component if the submitter is familiar with the abbreviation rules and convention. Proposed abbreviations are subject to review and change by the GTC.
* **Component Type** - The type of component. All repeating groups must be a component. A repeating group is a BlockRepeating component. If the component is not itself a repeating group (it can contain references to other components that are repeating groups), the component is a Block.
* **Category** - Assign each message to a Category. If a new category is required for the message, you must also complete the Category section.
* **Action** - Indicate whether the component table is for a new component being proposed, or a change to an existing component. Change to an existing component may include change to component synopsis or elaboration, addition of new or existing field(s), addition of new or existing component(s), etc.
* **Component Synopsis** - Required short description summarizing the purpose and function of the component.
* **Component Elaboration** - Optional detailed description of the message behavior.

|  |  |  |
| --- | --- | --- |
| To be completed at the time of the proposal – all information provided will be included in the repository | | |
| Component Name | | SecAltIDGrp |
| Component Abbreviated Name (for FIXML) | | AID |
| Component Type | | \_X\_ Block Repeating \_\_\_ Block |
| Category | | Common |
| Action | | \_\_New \_X\_Change |
| Component Synopsis  Required, short, one or two paragraph description of the component. | *(no change)* | |
| Component Elaboration  Optional longer description of the component usage | *(no change)* | |
| To be finalized by FPL Technical Office | | |
| Repository Component ID 2071 | |  |

A starter component format table is provided below. If you are starting a gap analysis that modifies an existing component, you may want to cut and paste the component from the FIX Protocol specification or request from the GTC (via the FPL Program Office) that a gap analysis template be provided that is pre-populated with the component tables from the most current Repository.

Clearly highlight changes and additions by highlighting the row yellow. For new fields, use "TBD" for tag numbers, and assign field names and appropriate definitions. Identify enumerations. Include new message types being introduced without a MsgType assigned.

Include any message usage rules as the "preamble" to the message table for that message. This is particularly important when an existing message type is used differently in different asset classes or business workflow models. This can be described in the Message Elaboration.

* **Tag** - The FIX tag number for an existing field. If adding a new field, insert "TBD".
* **FieldName** - The FIX tag name for an existing field. If adding a new field, provide a field name that reasonably indicates what the field is for.
* **Req’d** - Indicates whether the field is required or not in the component. For an existing FIX component, include the "Req'd" value based on the protocol version (e.g. FIX 4.4). For a new FIX component, the "Req'd" value is up to the message designer.
* **IRC** - Indicates whether a component reference is to be inlined in FIXML. This column usually can be ignored and is seldom part of an initial proposal since inlining a component reference is an implementation decision made by the GTC group.
* **Action** - Indicates whether a field was added (i.e. add a new field or add an existing field to an existing message), changed (i.e. a change to an existing field in the existing message), or deprecated (i.e. remove an existing field in the existing message). Leave this column blank if there is no change to definition, usage, or enumeration values for existing fields.

**ADD** - Add a referenced existing field to an existing component.

**NEW** - Add a new referenced field to an existing or a new component being created as part of this proposal to the message.

**DEPRECATE** - Indicates an existing field in an existing component as being unsupported. Deprecating of a field from an existing component does not mean the field is deprecated from the standard as it may still be used elsewhere.

**REMOVE** - Indicates the referenced existing field is to be removed from the existing component.

**CHANGE** - Change just the field usage comment in the component, Req’d, or ICR associated with a field reference in the existing component.

* **Mappings and Usage Comments** (blue heading) - This column can be used to document analysis notes, specific usage or mappings of the field to the business requirements, and indicating whether any new enumerations are being added to an existing field (which are to be listed in the Data Dictionary).
* **FIX Spec Comment** - Contains the existing usage text of the field for the existing component. For New fields to be added to an existing component this column contains any component specific usage comments for the field within the context of the component - *this usage comment should not be a duplicate of the new field's definition description used in the Data Dictionary.* For new components this contains any component specific usage comments for the field - *this usage comment should not be a duplicate of the new field's definition description used in the Data Dictionary*. These usage comments will be part of component table in the FIX Specification.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Component FIXML Abbreviation: <*SecAltID*> | | | | | | |
| *Tag* | *Field Name* | | *Req'd* | *Action* | *Mappings and Usage Comments* | *FIX Spec Comments* |
| 454 | NoSecurityAltID | |  |  |  |  |
| 🡪 | 455 | SecurityAltID |  |  |  |  |
| 🡪 | 456 | SecurityAltIDSource |  |  |  |  |
| 🡪 | TBD | SymbolPositionNumber | N | NEW |  |  |
| </*SecAltID*> | | | | | | |

## Component LegSecAltIDGrp

|  |  |  |
| --- | --- | --- |
| To be completed at the time of the proposal – all information provided will be included in the repository | | |
| Component Name | | LegSecAltIDGrp |
| Component Abbreviated Name (for FIXML) | | LegAID |
| Component Type | | \_X\_ Block Repeating \_\_\_ Block |
| Category | | Common |
| Action | | \_\_New \_X\_Change |
| Component Synopsis  Required, short, one or two paragraph description of the component. | *(no change)* | |
| Component Elaboration  Optional longer description of the component usage | *(no change)* | |
| To be finalized by FPL Technical Office | | |
| Repository Component ID 2072 | |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Component FIXML Abbreviation: <*LegSecAltIDGrp*> | | | | | | |
| *Tag* | *Field Name* | | *Req'd* | *Action* | *Mappings and Usage Comments* | *FIX Spec Comments* |
| 604 | NoLegSecurityAltID | |  |  |  |  |
| 🡪 | 605 | LegSecurityAltID |  |  |  |  |
| 🡪 | 606 | LegSecurityAltIDSource |  |  |  |  |
| 🡪 | TBD | LegSymbolPositionNumber | N | NEW |  |  |
| </*LegSecAltIDGrp*> | | | | | | |

## Component UndSecAltIDGrp

|  |  |  |
| --- | --- | --- |
| To be completed at the time of the proposal – all information provided will be included in the repository | | |
| Component Name | | UndSecAltIDGrp |
| Component Abbreviated Name (for FIXML) | | UndAID |
| Component Type | | \_X\_ Block Repeating \_\_\_ Block |
| Category | | Common |
| Action | | \_\_New \_X\_Change |
| Component Synopsis  Required, short, one or two paragraph description of the component. | *(no change)* | |
| Component Elaboration  Optional longer description of the component usage | *(no change)* | |
| To be finalized by FPL Technical Office | | |
| Repository Component ID 2073 | |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Component FIXML Abbreviation: <*UndSecAltIDGrp*> | | | | | | |
| *Tag* | *Field Name* | | *Req'd* | *Action* | *Mappings and Usage Comments* | *FIX Spec Comments* |
| 457 | NoUnderlyingSecurityAltID | |  |  |  |  |
| 🡪 | 458 | UnderlyingSecurityAltID |  |  |  |  |
| 🡪 | 459 | UnderlyingSecurityAltIDSource |  |  |  |  |
| 🡪 | TBD | UnderlyingSymbolPositionNumber | N | NEW |  |  |
| </*UndSecAltIDGrp*> | | | | | | |

# Category Changes

This chapter is used to add or modify FIX Categories. If your gap analysis proposal does not add or change categories, ignore this section during the initial proposal. If, during review, a change to an existing category or a new category is determined to be required, complete this section.

* **Category Name** - The category or repository name.
* **Section** - The section for the category. A message category can exist only in one section of the FIX Specification.
* **Category Synopsis** - Required short description summarizing the purpose and function of the component.
* **Category Elaboration** - Optional detailed description of the message behavior.

NONE

# Appendix A - Data Dictionary

The Data Dictionary table must be filled in for all new fields being proposed and all existing fields where changes are being proposed. Each row, representing a field, must identify the requested action of “new”, "add", "change", or "deprecate" for each field.

For new fields provide the data type for each field, the field definition, along with any enumerations related to the field. New fields will use "TBD" in the Tag column. For existing fields, document the proposed additions and changes and highlighting the change (e.g. to the description, new enumerations being added, etc.).

List new fields at the top of the table, followed by fields to be deprecated, and then fields to be changed.

* **Tag** - Order all new fields at the top of the table. The "Tag" column should be "TBD" for the new fields. For existing fields include the official tag number.
* **FieldName** – Field name – required for all fields including existing fields being changed and proposed.
* **Action** - indicates whether the field is to be added, changed, or deprecated in the data dictionary:

**NEW** - A new proposed field. Use "TBD" in Tag column. Identified in the "Add to/ Deprecate from Message type or Component block" column the message or component the new field is to be added to.

**ADD** - An existing field to be added to the component or message type identified in the "Add to/ Deprecate from Message type or Component block" column.

**DEPRECATE** - An existing field to be deprecated. If the deprecation is message specific (as oppose to deprecating the field from the entire specification), identify in the "Add to/ Deprecate from Message type or Component block" column the component or message from which the field is to be deprecated.

**CHANGE** - An existing field to be modified – modifications are limited to changing the Data Dictionary description or changing or adding new enumerations. A data type change requires strong business requirements justification to be documented as part of the proposal and will be reviewed in detail by the GTC.

* **Datatype** - The data type, e.g. int, Price, Boolean, etc. (See FIXimate for the complete list of FIX datatypes). Required for new fields; not required for existing fields, unless the proposal is to change the data type. See the list of data types in Volume 1 of the FIX Protocol specification.
* **Description -** A definition of the field. The description of the field should be sufficiently descriptive and meaningful but should be generic enough that the field can be reused. For specific message or component context based usage rules these should be described as field usage text within the message or component in which the field is included.

**Enumerations** - When a field requires enumerations, these are included within the **Description** column of the Data Dictionary table. When enumerations are to be defined for a new field, the field should be of *int* data type and the enumerated values be integers starting at 0 (zero). If the field is to have a default enumeration value that is implied by the omission of the field, the default value must be assigned the value 0 (zero). All enumerations must have a short description included that provides sufficient meaning for the enumeration value. A longer elaboration or description for the enumeration may also be included.

* **FIXML Abbreviation** - The abbreviation for the field when used in FIXML Schema. The submitter is not required to supply the abbreviation, but it will need to be entered in conjunction with the GTC before ratification of the proposal. If the submitter is familiar with the abbreviation rules or convention, the submitter is encouraged to supply a proposed FIXML abbreviation; however, this is subject to review and change by the GTC. If new terms require abbreviation, propose the abbreviation for the new term(s) in Appendix C.
* **Add to / Deprecate from Message type or Component block** - Identify the message types or component blocks in which to apply the Action for the field.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tag** | **FieldName** | **Action** | **Datatype** | **Description** | **FIXML Abbreviation** | **Add to / Deprecate from Message type or Component block** |
| TBD | CurrencyCodeSource | NEW | string | Identifies class or source of the Currency(15) value.  1 = CUSIP  2 = SEDOL  4 = ISIN  6 = ISO Currency Code (ISO 4217)  S = Financial Instrument Global Identifier (FIGI)  TBD = Digital Token Identifier (ISO 24165) | @Src | Please see *Table 1* and *Table 2* |
| TBD | LegCurrencyCodeSource | NEW | string | Identifies class or source of the LegCurrency(556) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | SettlCurrencyCodeSource | NEW | string | Identifies class or source of the SettlCurrency(120) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | LegSettlCurrencyCodeSource | NEW | string | Identifies class or source of the LegSettlCurrency(675) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | SideCurrencyCodeSource | NEW | string | Identifies class or source of the SideCurrency(1154) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | SideSettlCurrencyCodeSource | NEW | string | Identifies class or source of the SideSettlCurrency(1155) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | SettlementAmountCurrencyCodeSource | NEW | string | Identifies class or source of the SettlementAmountCurrency(1702) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | StrikeCurrencyCodeSource | NEW | string | Identifies class or source of the StrikeCurrency(947) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | UnitOfMeasureCurrencyCodeSource | NEW | string | Identifies class or source of the UnitOfMeasureCurrency(1716) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | PriceUnitOfMeasureCurrencyCodeSource | NEW | string | Identifies class or source of the PriceUnitOfMeasureCurrency(1717) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | PriceQuoteCurrencyCodeSource | NEW | string | Identifies class or source of the PriceQuoteCurrency(1524) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | LegStrikeCurrencyCodeSource | NEW | string | Identifies class or source of the LegStrikeCurrency(942) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | LegUnitOfMeasureCurrencyCodeSource | NEW | string | Identifies class or source of the LegUnitOfMeasureCurrency(1720) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | LegPriceUnitOfMeasureCurrencyCodeSource | NEW | string | Identifies class or source of the LegPriceUnitOfMeasureCurrency(1721) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | LegPriceQuoteCurrencyCodeSource | NEW | string | Identifies class or source of the LegPriceQuoteCurrency(1528) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | DerivativeStrikeCurrencyCodeSource | NEW | string | Identifies class or source of the DerivativeStrikeCurrency(1262) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | DerivativeUnitOfMeasureCurrencyCodeSource | NEW | string | Identifies class or source of the DerivativeUnitOfMeasureCurrency(1722) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | DerivativePriceUnitOfMeasureCurrencyCodeSource | NEW | string | Identifies class or source of the DerivativePriceUnitOfMeasureCurrency(1723) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | DerivativePriceQuoteCurrencyCodeSource | NEW | string | Identifies class or source of the DerivativePriceQuoteCurrency(1576) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | UnderlyingCurrencyCodeSource | NEW | string | Identifies class or source of the UnderlyingCurrency(318) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | UnderlyingStrikeCurrencyCodeSource | NEW | string | Identifies class or source of the UnderlyingStrikeCurrency(941) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | UnderlyingUnitOfMeasureCurrencyCodeSource | NEW | string | Identifies class or source of the UnderlyingUnitOfMeasureCurrency(1718)  value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | UnderlyingPriceUnitOfMeasureCurrencyCodeSource | NEW | string | Identifies class or source of the UnderlyingPriceUnitOfMeasureCurrency(1719) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | UnderlyingPriceQuoteCurrencyCodeSource | NEW | string | Identifies class or source of the UnderlyingPriceQuoteCurrency(1526) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | UnderlyingNotionalCurrencyCodeSource | NEW | string | Identifies class or source of the UnderlyingNotionalCurrency(2615) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | CommCurrencyCodeSource | NEW | string | Identifies class or source of the CommCurrency(479) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | CommissionCurrencyCodeSource | NEW | string | Identifies class or source of the CommissionCurrency(2643) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | CommissionUnitOfMeasureCurrencyCodeSource | NEW | string | Identifies class or source of the CommissionUnitOfMeasureCurrency(2645) value.  (*Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | AllocCommissionCurrencyCodeSource | NEW | string | Identifies class or source of the AllocCommissionCurrency(2657)  value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | AllocCommissionUnitOfMeasureCurrencyCodeSource | NEW | string | Identifies class or source of the AllocCommissionUnitOfMeasureCurrency(2659) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | AllocSettlCurrencyCodeSource | NEW | string | Identifies class or source of the AllocSettlCurrency(736) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | LegAllocSettlCurrencyCodeSource | NEW | string | Identifies class or source of the LegAllocSettlCurrency(1367) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | CollateralCurrencyCodeSource | NEW | string | Identifies class or source of the CollateralCurrency(1705) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | SideCollateralCurrencyCodeSource | NEW | string | Identifies class or source of the SideCollateralCurrency(2695) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | CollateralReinvestmentCurrencyCodeSource | NEW | string | Identifies class or source of the CollateralReinvestmentCurrency(2843) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | SideCollateralReinvestmentCurrencyCodeSource | NEW | string | Identifies class or source of the SideCollateralReinvestmentCurrency(2866) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | TradeAllocCurrencyCodeSource | NEW | string | Identifies class or source of the TradeAllocCurrency(1847) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | TradingCurrencyCodeSource | NEW | string | Identifies class or source of the TradingCurrency(1245) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | LimitAmtCurrencyCodeSource | NEW | string | Identifies class or source of the LimitAmtCurrency(1634) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | PosQtyUnitOfMeasureCurrencyCodeSource | NEW | string | Identifies class or source of the PosQtyUnitOfMeasureCurrency(1835) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | PositionCurrencyCodeSource | NEW | string | Identifies class or source of the PositionCurrency(1055) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | LegPosCurrencyCodeSource | NEW | string | Identifies class or source of the LegPosCurrency(1589) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | RiskLimitCurrencyCodeSource | NEW | string | Identifies class or source of the RiskLimitCurrency(1532) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | EntitlementAttribCurrencyCodeSource | NEW | string | Identifies class or source of the EntitlementAttribCurrency(1781) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | ComplexOptPayoutCurrencyCodeSource | NEW | string | Identifies class or source of the ComplexOptPayoutCurrency(2122) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | ComplexEventCurrencyOneCodeSource | NEW | string | Identifies class or source of the ComplexEventCurrencyOne(2124) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | ComplexEventCurrencyTwoCodeSource | NEW | string | Identifies class or source of the ComplexEventCurrencyTwo(2125) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | LegComplexOptPayoutCurrencyCodeSource | NEW | string | Identifies class or source of the LegComplexOptPayoutCurrency(2226) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | LegComplexEventCurrencyOneCodeSource | NEW | string | Identifies class or source of the LegComplexEventCurrencyOne(2233) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | LegComplexEventCurrencyTwoCodeSource | NEW | string | Identifies class or source of the LegComplexEventCurrencyTwo(2234) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | UnderlyingComplexOptPayoutCurrencyCodeSource | NEW | string | Identifies class or source of the UnderlyingComplexOptPayoutCurrency(2266) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | UnderlyingComplexEventCurrencyOneCodeSource | NEW | string | Identifies class or source of the UnderlyingComplexEventCurrencyOne(2268) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | UnderlyingComplexEventCurrencyTwoCodeSource | NEW | string | Identifies class or source of the UnderlyingComplexEventCurrencyTwo(2269) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | BenchmarkCurveCurrencyCodeSource | NEW | string | Identifies class or source of the BenchmarkCurveCurrency(220) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | LegBenchmarkCurveCurrencyCodeSource | NEW | string | Identifies class or source of the LegBenchmarkCurveCurrency(676) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | AgreementCurrencyCodeSource | NEW | string | Identifies class or source of the AgreementCurrency(918) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | LegAgreementCurrencyCodeSource | NEW | string | Identifies class or source of the LegAgreementCurrency(2495) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | FundingSourceCurrency | NEW | string | Identifies class or source of the FundingSourceCurrency(2847) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | PayCollectCurrencyCodeSource | NEW | string | Identifies class or source of the PayCollectCurrency(1709) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | PostTradePaymentCurrencyCodeSource | NEW | string | Identifies class or source of the PostTradePaymentCurrency(2818) value.  *(Uses enums from CurrencyCodeSource(TBD))* | @Src | Please see *Table 1* and *Table 2* |
| TBD | SymbolPositionNumber | NEW | int | Reference to the first or second currency or digital asset in Symbol(55) for FX-style trading.  Conditionally required when one or both symbols in Symbol(55) represent a digital asset. | @SymPosNum | Add to SecAltIDGrp component |
| TBD | LegSymbolPositionNumber | NEW | int | Reference to the first or second currency or digital asset in LegSymbol(600) for FX-style trading.  Conditionally required when one or both symbols in LegSymbol(600) represent a digital asset. | @SymPosNum | Add to LegSecAltIDGrp component |
| TBD | UnderlyingSymbolPositionNumber | NEW | int | Reference to the first or second currency or digital asset in UnderlyingSymbol(311) for FX-style trading.  Conditionally required when one or both symbols in UnderlyingSymbol(311) represent a digital asset. | @SymPosNum | Add to UndSecAltIDGrp component |
| 15 | Currency | CHANGE | Currency | *Correct/append description and add elaboration:*  Identifies currency used for price or quantity fields, depending on the asset class being traded. CurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent.  Absence of this field is interpreted as the default for the security. It is recommended that systems provide the currency value whenever possible.  ~~See "Appendix 6-A: Valid Currency Codes" for information on obtaining valid values.~~  [Elaboration: For securities trading and  digital assets traded securities-style, identifies the currency used to denote the price. Absence of this field is interpreted as the default for the security. For Foreign Exchange (FX) and digital assets traded FX-style, identifies the dealt currency used to denominate the quantity related field(s).] | @Ccy | Please see table 1 and 2 |
| 22 | SecurityIDSource | CHANGE | String | Identifies class or source of the SecurityID(48) value.  *Change enumeration:*  *6 = ISO Currency Code (ISO 4217)*  *Add enumeration:*  TBD = Digital Token Identifier (ISO 24165) | @Src |  |
| 167 | SecurityType | CHANGE | String | Indicates type of security.  *Add enumeration to* ***Other****:*  DIGITAL = Digital Asset  [Elaboration: Asset that exists only in digital form or which is the digital representation of another asset (Source: ISO 24165 - Terms and Definitions).] | @SecTyp |  |
| 220 | BenchmarkCurveCurrency | CHANGE | Currency | *Correct and append description:*  Specifies currency used for benchmark curve. ~~See "Appendix 6-A; Valid Currency Codes" for information in obtaining valid values.~~  BenchmarkCurveCurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 318 | UnderlyingCurrency | CHANGE | Currency | *Correct description:*  Underlying security's ~~C~~currency.  ~~See Currency (5) field for description and valid values.~~ | @Ccy |  |
| 479 | CommCurrency | CHANGE | Currency | *Correct and append description:*  Specifies currency to be use for Commission(12) if the Commission currency is different from the Deal Currency. ~~See "Appendix 6-A; Valid Currency Codes".~~  CommCurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 803 | PartySubIDType | CHANGE | int | Type of PartySubID(523) value.  *Add enumeration:*  TBD = Wallet Identifier  [Elaboration: TBD] | @Typ |  |
| 1634 | LimitAmtCurrency | CHANGE | Currency | *Correct description:*  Indicates the currency that the limit amount is specified in. ~~See Currency(15) for additional description and valid values.~~ | @Ccy |  |
| 2122 | ComplexOptPayoutCurrency | CHANGE | Currency | *Correct and append description:*  Specifies the currency of the payout amount. ~~Uses ISO 4217 currency codes.~~  ComplexOptPayoutCurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2124 | ComplexEventCurrencyOne | CHANGE | Currency | *Correct and append description:*  Specifies the first or only reference currency of the trade. ~~Uses ISO 4217 currency codes.~~  ComplexEventCurrencyOneCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2125 | ComplexEventCurrencyTwo | CHANGE | Currency | *Correct and append description:*  Specifies the second reference currency of the trade. ~~Uses ISO 4217 currency codes.~~  ComplexEventCurrencyTwoCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2226 | LegComplexOptPayoutCurrency | CHANGE | Currency | *Correct and append description:*  Specifies the currency of the payout amount. ~~Uses ISO 4217 currency codes.~~  LegComplexOptPayoutCurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2233 | LegComplexEventCurrencyOne | CHANGE | Currency | *Correct and append description:*  Specifies the first or only reference currency of the trade. ~~Uses ISO 4217 currency codes.~~  LegComplexEventCurrencyOneCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2234 | LegComplexEventCurrencyTwo | CHANGE | Currency | *Correct and append description:*  Specifies the second reference currency of the trade. ~~Uses ISO 4217 currency codes.~~  LegComplexEventCurrencyTwoCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2266 | UnderlyingComplexOptPayoutCurrency | CHANGE | Currency | *Correct and append description:*  Specifies the currency of the payout amount. ~~Uses ISO 4217 currency codes.~~  UnderlyingComplexOptPayoutCurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2268 | UnderlyingComplexEventCurrencyOne | CHANGE | Currency | *Correct and append description:*  Specifies the first or only reference currency of the trade. ~~Uses ISO 4217 currency codes.~~  UnderlyingComplexEventCurrencyOneCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2269 | UnderlyingComplexEventCurrencyTwo | CHANGE | Currency | *Correct and append description:*  Specifies the second reference currency of the trade. ~~Uses ISO 4217 currency codes.~~  UnderlyingComplexEventCurrencyTwoCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2615 | UnderlyingNotionalCurrency | CHANGE | Currency | *Correct and append description:*  Specifies the currency denomination of the notional value. ~~Uses ISO 4217 currency codes.~~  UnderlyingNotionalCurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2643 | CommissionCurrency | CHANGE | Currency | *Correct and append description:*  Specifies the currency denomination of the commission amount if different from the trade's currency. ~~Uses ISO 4217 currency codes.~~  CommissionCurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2657 | AllocCommissionCurrency | CHANGE | Currency | *Correct and append description:*  Specifies the currency denomination of the commission amount if different from the trade's currency. ~~Uses ISO 4217 currency codes.~~  AllocCommissionCurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2695 | SideCollateralCurrency | CHANGE | Currency | *Correct and append description:*  Specifies the currency if the collateral; optional, defaults to settlement currency if not specified. ~~Uses ISO 4217 currency codes.~~  SideCollateralCurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2818 | PostTradePaymentCurrency | CHANGE | Currency | *Correct and append description:*  Specifies the currency in which PostTradePaymentAmount(2817) is denominated. ~~Uses ISO 4217 currency codes.~~  PostTradePaymentCurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2843 | CollateralReinvestmentCurrency | CHANGE | Currency | *Correct and append description:*  The currency denomination of the re-invested cash amount. ~~Uses ISO 4217 currency codes.~~  CollateralReinvesmentCurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2847 | FundingSourceCurrency | CHANGE | Currency | *Correct and append description:*  Currency denomination of the market value of the funding source. ~~Uses ISO 4217 currency codes.~~  FundingSourceCurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |
| 2866 | SideCollateralReinvestmentCurrency | CHANGE | Currency | *Correct and append description:*  The currency denomination of the re-invested cash amount. ~~Uses ISO 4217 currency codes.~~  SideCollateralReinvestmentCurrencyCodeSource(TBD) may be used to disambiguate the code source scheme used, and ISO 4217 is the default scheme if absent. | @Ccy |  |

# Appendix B - Glossary Entries

This section, if included, should contain a table with terminology to be included in the FIX specification Glossary in Volume 1. These are usually business terms that are defined to help readers understand the relevant space for the proposal.

* **Term** - The business term.
* **Definition** - The definition of the term. If a term has different definitions in different contexts or for different asset types, include and identify fully these differing definitions. If the definition is copied or paraphrased from a source, identify the source in parentheses after the definition.
* **Field where used** - Identifies the FIX field name for the field where this term is used.

|  |  |  |
| --- | --- | --- |
| **Term** | **Definition** | **Field where used** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Appendix C - Abbreviations

When new fields, components, and messages are added to the FIX Specification, an abbreviated name that is primarily used for FIXML (at this time) must be created for them. Abbreviations are standardized within the FIX Specification. A list of abbreviations is maintained in the FIX Repository. You can access the current list of abbreviations via FIXimate on the FPL website. If abbreviations do not exist, use this table to define additional abbreviations required for your proposal. New abbreviations are subject to final approval of and may be changed by the GTC.

If you are not comfortable proposing new abbreviations, the "Proposed Abbreviations" can be omitted and the GTC will assign new abbreviations.

|  |  |  |
| --- | --- | --- |
| **Term** | **Proposed Abbreviation** | **Proposed Messages, Components, Fields where used** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Appendix D - Usage Examples

This is an optional section where the sub-committee or working group can provide whole or fragments of example FIX messages with actual or dummy data. These examples are useful for illustrating usage or rules specific to the business domain covered in the proposal.

**Fiat ccy vs. fiat ccy, “FX-style”**

|  | **Buying 1M EUR** | **Selling 1.4M USD** |
| --- | --- | --- |
| **ORDERS & EXECUTIONS** |  |  |
| **Symbol(55)** | EUR/USD | EUR/USD |
| **Side(54)** | Buy | Sell |
| **Direction - PriceType(423)** | 20 (Normal - CCY1 \* rate) | 21 (Inverse - CCY1 / Rate) |
| **ORDERS** |  |  |
| **Rate – Price(44)** | 1.4 | 1.4 |
| **Dealing Qty – OrderQty(38)** | 1000000 | 1400000 |
| **Dealing Ccy - Currency(15)** | EUR | USD |
| ***Optional unless settling in a third currency:*** |  |  |
| **SettlCurrency(120)** | USD | EUR |
| **EXECUTIONS** |  |  |
| **Rate – LastPx(31) and AvgPx(6)** | 1.4 | 1.4 |
| **Dealt Qty – LastQty(32)** | 1000000 | 1400000 |
| **Dealt Ccy - Currency(15)** | EUR | USD |
| **CalculatedCcyLastQty(1056)** | 1400000 | 1000000 |
| ***Optional unless settling in a third currency:*** |  |  |
| **SettlCurrency(120)** | USD | EUR |
| **SettlCurrAmt(119)** | 1400000 | 1000000 |

**Digital, “FX-Style”**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Buying 2.5 Bitcoin in USD** | **Buying $100,000 of Bitcoin**  ***(or selling $100,000 for Bitcoin)*** | **Buying 10 Ether in Bitcoin** |
| **ORDERS & EXECUTIONS** |  |  |  |
| **Symbol (55)** | BTC/USD | BTC/USD | ETH/BTC |
| **NoSecurityAltID(454)** | 2 | 2 | 2 |
| **SecurityAltID(455)** | 4H95J0R2X (Bitcoin’s DTI) | 4H95J0R2X (Bitcoin’s DTI) | X9J9K8725 (Ether’s DTI) |
| **SecurityAltIDSource(456)** | New value to represent DTI | New value to represent DTI | New value to represent DTI |
| ***SymbolPositionNumber(TBD)*** | *1* | *1* | *1* |
| **SecurityAltID(455)** | USD | USD | 4H95J0R2X (Bitcoin’s DTI) |
| **SecurityAltIDSource(456)** | 6 | 6 | New value to represent DTI |
| ***SymbolPositionNumber(TBD)*** | *2* | *2* | *2* |
| **Side(54)** | Buy | Sell (because we are selling USD) | Buy |
| **Direction - PriceType(423)** | 20 (Normal - CCY1 \* rate) | 21 (Inverse - CCY1 / rate) | 20 (Normal - CCY1 \* rate) |
| **ORDERS** |  |  |  |
| **Rate – Price(44)** | 40,000 (price of 1 BTC in USD) | 40,000 (price of 1 BTC in USD) | 0.075 (price of 1 ETH in BTC) |
| **Dealing Qty – OrderQty(38)** | 2.5 (in Bitcoin) | 100,000 (in USD) | 10 (in Ether) |
| **Dealing Ccy - Currency(15)** | 4H95J0R2X (Bitcoin) | USD | X9J9K8725 (Ether) |
| ***CurrencyCodeSource(TBD)*** | *TBD=Digital Token Identifier* | *6=ISO Currency Code* | *TBD=Digital Token Identifier* |
| ***Optional unless settling in a third currency:*** |  |  |  |
| **SettlCurrency(120)** | USD | 4H95J0R2X (Bitcoin’s DTI) | 4H95J0R2X (Bitcoin’s DTI) |
| ***SettlCurrencyCodeSource(TBD)*** | *6=ISO Currency Code* | *TBD=Digital Token Identifier* | *TBD=Digital Token Identifier* |
| **EXECUTIONS** |  |  |  |
| **Rate – LastPx(31) and AvgPx(6)** | 40,000 (price of 1 BTC in USD) | 40,000 (price of 1 BTC in USD) | 0.075 (price of 1 ETH in BTC) |
| **Dealt Qty – LastQty(32)** | 2.5 (in Bitcoin) | 100,000 (in USD) | 10 (in Ether) |
| **Dealt Ccy - Currency(15)** | 4H95J0R2X (Bitcoin) | USD | X9J9K8725 (Ether) |
| ***CurrencyCodeSource(TBD)*** | *TBD=Digital Token Identifier* | *6=ISO Currency Code* | *TBD=Digital Token Identifier* |
| **CalculatedCcyLastQty(1056)** | 100,000 (in USD) | 2.5 (in Bitcoin) | 0.75 (in Bitcoin) |
| ***Optional unless settling in a third currency:*** |  |  |  |
| **SettlCurrency(120)** | USD | 4H95J0R2X (Bitcoin) | 4H95J0R2X (Bitcoin’s DTI) |
| ***SettlCurrencyCodeSource(TBD)*** | *6=ISO Currency Code* | *TBD=Digital Token Identifier* | *TBD=Digital Token Identifier* |
| **SettlCurrAmt(119)** | 100,000 (in USD) | 2.5 (in Bitcoin) | 0.75 (in Bitcoin) |

**Digital, “Securities-Style”**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Buying 250 MSFT in USD** | **Buying 5 ‘crypto-Apple shares’ in USD** | **Buying $100,000 of Bitcoin** | **Buying 10 Ether in Bitcoin** |
| **ORDERS AND EXECUTIONS** |  |  |  |  |
| **Symbol(55)** | MSFT | CAAP | BTC | ETH |
| **SecurityID(48)** | 594918104 | A1B2C3D4E | 4H95J0R2X (Bitcoin’s DTI) | X9J9K8725 (Ether’s DTI) |
| **SecurityIDSource(22)** | *1 (CUSIP)* | *TBD=Digital Token Identifier* | *TBD=Digital Token Identifier* | *TBD=Digital Token Identifier* |
| **Side(54)** | Buy | Buy | Buy | Buy |
| **ORDERS** |  |  |  |  |
| **Price – Price(44)** | 311 | 500 | 40,000 | 0.075 (price of 1 ETH in BTC) |
| **Quantity – OrderQty(38)** | 250 | 5 |  | 10 |
| **Quantity – CashOrderQty(152)** |  |  | 100,000 |  |
| **Dealing Ccy - Currency(15)** | USD | USD | USD | 4H95J0R2X (Bitcoin) |
| ***CurrencyCodeSource(TBD)*** | *6=ISO Currency Code* | *6=ISO Currency Code* | *6=ISO Currency Code* | *TBD=Digital Token Identifier* |
| ***Optional unless settling in a third currency:*** |  |  |  |  |
| **SettlCurrency(120)** | USD | USD | 4H95J0R2X (Bitcoin) | 4H95J0R2X (Bitcoin) |
| ***SettlCurrencyCodeSource(TBD)*** | *6=ISO Currency Code* | *6=ISO Currency Code* | *TBD=Digital Token Identifier* | *TBD=Digital Token Identifier* |
| **EXECUTIONS** |  |  |  |  |
| **Price – LastPx(31) and AvgPx(6)** | 311 | 500 | 40,000 | 0.075 (price of 1 ETH in BTC) |
| **Quantity – LastQty(32)** | 250 | 5 | 2.5 | 10 |
| **Dealt Ccy - Currency(15)** | USD | USD | USD | 4H95J0R2X (Bitcoin) |
| ***CurrencyCodeSource(TBD)*** | *6=ISO Currency Code* | *6=ISO Currency Code* | *6=ISO Currency Code* | *TBD=Digital Token Identifier* |
| ***Optional unless settling in a third currency:*** |  |  |  |  |
| **SettlCurrency(120)** | USD | USD | 4H95J0R2X (Bitcoin) | 4H95J0R2X (Bitcoin) |
| ***SettlCurrencyCodeSource(TBD)*** | *6=ISO Currency Code* | *6=ISO Currency Code* | *TBD=Digital Token Identifier* | *TBD=Digital Token Identifier* |
| **SettlCurrAmt(119) – on ERs** | 77750 | 2500 | 2.5 (in Bitcoin) | 0.75 |