# A NEW APPROACH TO STP: FIX 4.4 – ISO15022

| Version | Author | Date | Notes |
|---|---|---|---|
| 0.1 | Simon Leighton-Porter | 22 May 03 | First draft |
| 0.2 | Simon Leighton-Porter | 26 May 03 | Addition of parties information |
| 0.3 | Simon Leighton-Porter | 27 May 03 | Minor |
| 0.4 | Simon Leighton-Porter | 28 May 03 | Minor |

*Acknowledgement: much of the text of this document has been taken directly or paraphrased from the FIX 4.4 specification published by FPL.*

# 1. INTRODUCTION AND EXECUTIVE SUMMARY

This paper addresses a number of fundamental aims affecting the global securities industry:

1. Achieving efficient cross-border STP at minimum cost to all industry participants.
2. Making best use of the new post-execution functionality within FIX 4.4
3. Defining and agreeing the data standards to be used in the new environment so that FIX can be mapped unambiguously to SWIFT in ISO15022 format.

The method proposed relies on the use of FIX for all post-execution messaging up to, but not including, settlement messages. It provides for real-time message enrichment with SSIs and net money. In addition, it proposes a method for identifying legal counterparties and their funds without the use of proprietary databases. These capabilities give the industry a viable, low-cost STP model.

The paper is broken into 4 main sections:

1. A high-level description of the proposed flow.
2. Overview of FIX component blocks and their uses.
3. FIX to SWIFT mapping
4. Legal entity and fund identifiers

**Achieving the reality**

The most pressing task is to define how FIX settlement information should be used for each combination of market, instrument and settlement location.

**Suggested method**

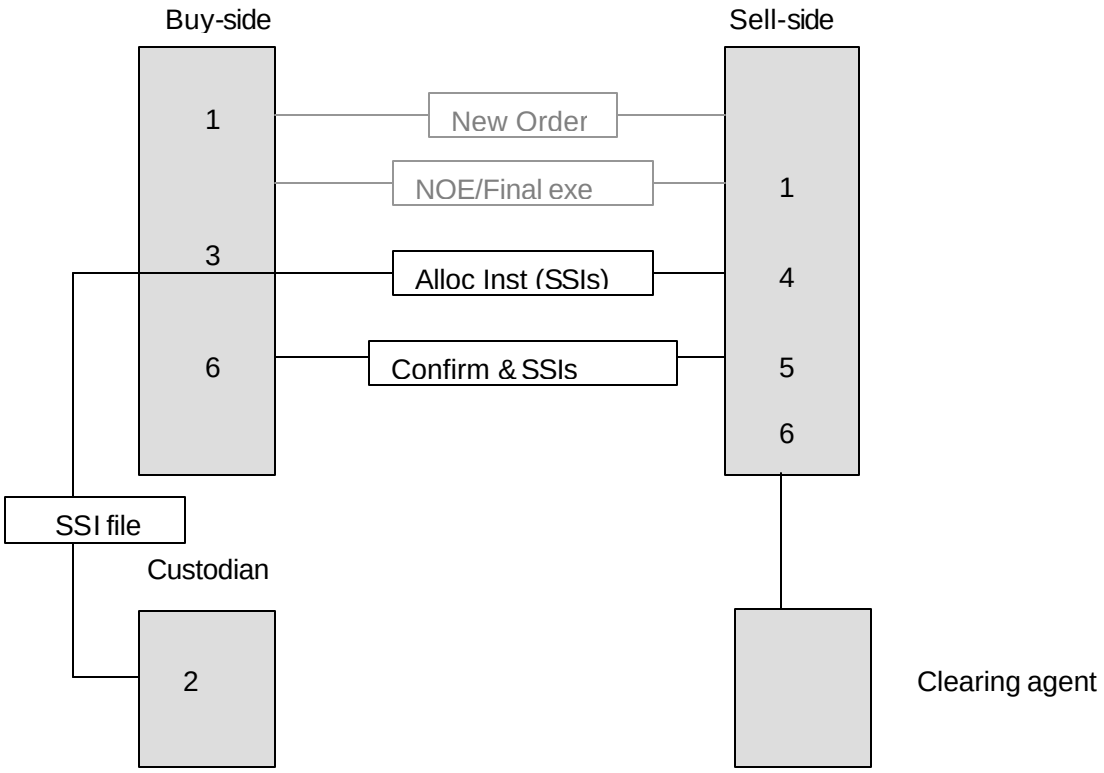
1. Start with the ideal MT54x for each combination (market, instrument, PSET).
2. From the MT54x, define the FIX message content and format.
3. From 1 and 2 above, define the format and content of the FIX Settlement Instruction message to support the proposed flow.

These steps will form the basis of a project to define data standards.

# 2. PROPOSED POST-EXECUTION FLOW

## *2.1 Overview*

1. Order and NOE/final execution report can take place via FIX or any other method.
2. Custodian sends periodic SSI update file to buy-side database.
3. Buy-side sends Allocation Instruction. Allocation includes total allocated quantity plus individual allocation quantities thus acting as both "block" and allocation. Allocations are enriched with SSIs from OMS database. Allocations may also contain net money calculations.
4. Sell-side matches SSIs for settlement channel compatibility and net money (if applicable).
5. Sell-side sends confirms enriched with both parties' SSIs and net money.
6. Buy-side matches complete trade and if match successful, instructs custodian and sends affirmation to sell-side (Confirmation ACK: type "accepted"). On receipt of affirmation, sell-side instructs its clearing agent.




## *2.2 Implementation*

1. The proposed flow is NOT dependent on FIX being used for order routing and the transmission of the NOE/final execution report.
2. Buy-side firms will need a database within their systems to hold their SSI information. This data will be maintained by the custodian and periodically updated. It will need to be available to the OMS in real time and will be used to

enrich allocation instruction messages. Custodians, OMS vendors/database providers and buy-sides must agree a common format and transmission method for the periodic SSI update file: the FIX 4.4 Settlement Instruction message appears to be a good candidate *but rigorous adoption of the ISO15022 format is essential.* The additional cost and effort of maintaining and handling this data will be more than offset by the reduction in manual investigation and repair work in sell-side, buy-side and custodians' operations departments.

3. The existence of an SSI database linked to the buy-side OMS will allow real-time enrichment of allocations. As an alternative to proprietary data sources, the buy-side firm may use the Legal Entity Identifier (LEI) and Fund LEI as outlined in appendix 2 (Legal Entity and Fund Data) below. Buy-side firms can choose one of the following options:
    a. No SSIs on Allocation Instruction Message: sell-side to use internal database or Alert/SID to look up SSIs.
    b. PSET only: sell-side will use PSET to find SSI in internal database and to perform settlement channel matching.
    c. Full SSI enrichment from the SSI database.

    Use of PSET or full SSI enrichment will allow sell-side firms to do settlement channel matching and thus avoid rejected confirmations and associated settlement issues. Ideally, net money calculations should also be included on the allocation. By providing both net money and PSET information on the allocation, potential mis-matches will be caught and repaired earlier. The sell-side in turn will populate their Confirmations with SSIs and net money. To gain maximum value from the process, buy-side firms should consider developing the capability to match the sell-side's information against the data sent on the Allocation Instruction.

    The use of the ISO15022 standard for settlement instructions will allow users to populate their SWIFT messages more easily, and the provision of just-in-time enrichment will allow both parties to the trade to do away with having to maintain a database of each-other's SSIs: the only SSIs they will have to maintain are their own. In order to achieve high levels of matching and automation, it is vital that when settlement instructions are transmitted via FIX that they can be unambiguously translated into ISO15022 format for inclusion in MT54x messages.
4. The adoption of LEIs (see appendix 1) will allow the use of post-execution FIX messaging by firms who do not currently subscribe to proprietary data sources.

**Notes on use of Settlement Location (PSET) and Trade Matching**

One of the strengths of the FIX 4.4 post-execution process is the ability to enrich messages with PSET or full settlement details. This will allow the sell-side to match the buy-side's PSET for "settlement channel compatibility" prior to the confirmation process. The sell-side will compare the PSET on the buy-side's Allocation Instruction with their default PSET for the security in question and, if the match is not exact, they will use their own proprietary logic to determine whether or not to support a "bridge" between the 2 PSETs. All participants are strongly encouraged to use the BIC for sending PSET information. This matching logic closely mimics that proposed by the GSTPA model and has the advantage of alerting parties to potential settlement issues well before instructions are sent to the market. For similar reasons, buy-side firms are encouraged to include net money calculations on their allocations.

# 3. FIX SETTLEMENT AND PARTIES INFORMATION

This section covers the usage of the following component blocks within the Allocation Instruction and Confirmation messages in FIX version 4.4.

# 4. Component Blocks

The use of Component Blocks is a flexible mechanism that allows new "roles" or types of firm identification to be specified without a corresponding increase in the number of FIX fields.

## *4.1 Parties*

In simple terms, the Parties block is used to describe those parties which are common to the entire message (in the case of the Allocation Instruction) or for those messages where there is no requirement to nest party information such as the Confirmation message. In addition, the PartyRole field allows users to specify "what" is being described or the role the party plays in the process. Similarly, the format or "source" of the identifier is shown via the PartyIDSource field. The PartySubID may be optionally used to provide an further level of subdivision

## *4.2 NestedParties*

The NestedParties block is used to describe parties that are relevant to discrete items on a message, such as individual allocations, where this information may differ from that of the "block" or where it varies from item to item. To illustrate its use, consider the Confirmation message which has no need for a NestedParties group because each Confirmation message is a unique message in itself and not one of many nested items such as the individual allocations within an Allocation Instruction message.

## *4.3 SettlInstructions and SettlParties*

The role of the SettlInstructions component block is self-explanatory and it can contain any number of SettlPartyIds that provide the detail on a settlement instruction. Its presence on an Allocation Instruction Message is optional. A confirmation can contain both sell-side and buy-side settlement parties and can be used for DVP/RVP, free and tri-party settlement. This block will be the main focus of the FIX-SWIFT mapping effort.

# 5. Location and Content of Component Blocks

## *5.1 Allocation Instruction*

This shows an Allocation Instruction message from which most of the fields have been removed for clarity. It shows the Parties block which holds information relevant to the entire message; the NestedParties block which holds information relevant to each individual allocation, and the SettlInstructions block, again holding information on an individual allocation level. Repeating within the SettlInstructions block is the SettlParties block. The component blocks are shown in **bold**.

| *Tag* | *Field Name* | *Req'd* | *Comments* |
|---|---|---|---|
| | *Standard Header* | Y | MsgType = J |
| 70 | AllocID | Y | Unique identifier for this allocation instruction message |
| Component block <Instrument> | | Y | Included for completeness only |
| **component block <Parties>** | | **N** | **The parties in this block must be common to the entire allocation** |
| *component block <YieldData>* | | N | Included for completeness only |
| 78 | NoAllocs | Y** | |
| → | *79 AllocAccount* | Y** | |
| → | ***component block <NestedParties>*** | **N** | **This block is used for party information that is specific to the individual allocation.** |
| → | *component block <CommissionData>* | N | Included for completeness only |
| → | *780 AllocSettlInstType* | N | Used to indicate whether settlement instructions are provided on this message, and if not, how they are to be derived.<br>Absence of this field implies use of default instructions. |
| → | ***component block <SettlInstructions>*** | **N** | **This block contains settlement instructions for this individual allocation only** |
| | *Standard Trailer* | Y | |

## *5.2 Confirmation*

As shown above, the Confirmation message has no need of a NestedParties block.

| *Tag* | *Field Name* | *Req'd* | *Comments* |
|---|---|---|---|
| | *Standard Header* | Y | MsgType = AK |
| 664 | ConfirmID | Y | Unique ID for this message |
| 650 | LegalConfirm | N | Denotes whether this message represents the legally binding confirmation<br>Required for ConfirmType = Confirmation |
| 665 | ConfirmStatus | Y | |
| **component block <Parties>** | | **N** | **Note that there is no need for a <NestedParties> block on the confirmation because the sell-side send an individual confirmation message in response to each allocation repeating within the Allocation Instruction message** |
| component block <Instrument> | | Y | |
| component block <YieldData> | | N | |
| **component block <SettlInstructions>** | | **N** | |
| component block <CommissionData> | | N | |
| | *Standard Trailer* | Y | |

## 6. Which Party and Where?

From a first reading of the FIX 4.4 specification, the 3 parties groups seem to overlap and there is a degree of flexibility over where such an attribute such as Place of Settlement (PSET) should be stated. To make this decision easier, all 3 groups share the same set of possible values and the correspondence between each of the groups is shown in the table below. (See volumes 1 and 6 of the FIX 4.4 specification for clarification if unclear)

| *Tag* | ***Party*** *Field Name* | *Tag* | ***Nested Party*** *Field Name* | *Tag* | ***Settlement Party*** *Field Name* |
|---|---|---|---|---|---|
| 453 | NoPartyIDs | 539 | NoNestedPartyIds | 781 | NoSettlPartyIDs |
| → | *448 PartyID* | → | *524 NestedPartyId* | → | *782 SettlPartyID* |
| → | *447 PartyIDSource* | → | *525 NestedPartyIdSource* | → | *783 SettlPartyIdSource* |
| → | *452 PartyRole* | → | *538 NestedPartyRole* | → | *784 SettlPartyRole* |
| → | *802 NoPartySubIDs* | → | *804 NoNestedPartySubIds* | → | *801 NoSettlPartySubIDs* |
| → | → *523 PartySubId* | → | → *545 NestedPartySubId* | → | → *785 SettlPartySubID* |
| → | → *803 PartySubIDType* | → | → *805 NestedPartySubIdType* | → | → *786 SettlPartySubIDType* |

## 7. PARTY

Parties are identified by a PartyID which in turn is "described" by the PartyRole, PartyIDSource and, optionally, by the PartySubID.

### *7.1 PartyRoles*

The PartyRole tells us "what" the PartyID being described is – a clearing firm, a custodian, an executing firm etc

The next table shows the PartyRole values that can be used in these 3 party blocks. The list is taken from the 4.4 spec. However, it should be noted that the meaning of some of these PartyRoles are ambiguous and do not map easily to ISO15022 standards.

| | **Party Role** | **Common Identification and Considerations Reference** |
|---|---|---|
| 1 | Executing Firm | See "Common PartyRole Identification for Firms" |
| 2 | Broker of Credit | See "Common PartyRole Identification for Firms" |
| 3 | Client ID | See "Common PartyRole Identification for Firms" |
| 4 | Clearing Firm | See "Common PartyRole Identification for Firms" |
| 5 | Investor ID | See "PartyRole Identification for Investor ID" |
| 6 | Introducing Firm | See "Common PartyRole Identification for Firms" |
| 7 | Entering Firm | See "Common PartyRole Identification for Firms" |
| 8 | Locate/Lending Firm (for short-sales) | See "Common PartyRole Identification for Firms" |
| 9 | Fund manager Client ID (for CIV) | See "Common PartyRole Identification for Firms" |
| 10 | Settlement Location | See "PartyRole Identification for Settlement Location" |
| 11 | Order Origination Trader (associated with Order Origination Firm – e.g. trader who initiates/submits the order) | See "Common PartyRole Identification for Traders" |
| 12 | Executing Trader (associated with Executing Firm - actually executes) | See "Common PartyRole Identification for Traders" |
| 13 | Order Origination Firm (e.g. buyside firm) | See "Common PartyRole Identification for Firms" |

| | Party Role | Common Identification and Considerations Reference |
|---|---|---|
| 14 | Giveup Clearing Firm (firm to which trade is given up) | See "Common PartyRole Identification for Firms" |
| 15 | Correspondent Clearing Firm | See "Common PartyRole Identification for Firms" |
| 16 | Executing System | See "PartyRole Identification for Execution System" |
| 17 | Contra Firm | See "Common PartyRole Identification for Firms" |
| 18 | Contra Clearing Firm | See "Common PartyRole Identification for Firms" |
| 19 | Sponsoring Firm | See "Common PartyRole Identification for Firms" |
| 20 | Underlying Contra Firm | See "Common PartyRole Identification for Firms" |
| 21 | Clearing Organization | See "Common PartyRole Identification for Firms" |
| 22 | Exchange | See "Common PartyRole Identification for Firms" |
| 24 | Customer Account | |
| 25 | Correspondent Clearing Organization | See "Common PartyRole Identification for Firms" |
| 26 | Correspondent Broker | See "Common PartyRole Identification for Firms" |
| 27 | Buyer/Seller (Receiver/Deliverer) | Value intended to be used in SettlParties component block (note these values correspond to ISO15022 settlement party categories) |
| 28 | Custodian | Value intended to be used in SettlParties component block (note these values correspond to ISO15022 settlement party categories) |
| 29 | Intermediary | Value intended to be used in SettlParties component block (note these values correspond to ISO15022 settlement party categories)<br>Note: it is possible to have multiple parties with this role in a SettlParties component block (intermediary 1, intermediary 2 etc.) in which case the PartySubID is used to distinguish between them. |
| 30 | Agent | Value intended to be used in SettlParties component block (note these values correspond to ISO15022 settlement party categories) |
| 31 | Sub custodian | Value intended to be used in SettlParties component block (note these values correspond to ISO15022 settlement party categories) |
| 32 | Beneficiary | Value intended to be used in SettlParties component block (note these values correspond to ISO15022 settlement party categories) |
| 33 | Interested Party | See "Common PartyRole Identification for Firms" |
| 34 | Regulatory body | See "Common PartyRole Identification for Firms" |
| 35 | Liquidity provider | See "Common PartyRole Identification for Firms" |

## *7.2 PartyIDSource*

The PartyIDSource tells us the format of the information provided in the PartyIDRole eg BIC, generally accepted market participant identifier, ISO country code etc

### 7.2.1 PartyIDSource for Firms

| PartyIDSource (447) | | PartyID (448) | PartySubID (523) |
|---|---|---|---|
| B | BIC (Bank Identification Code) | <<BIC Code Value>> | (optional) |
| C | Generally accepted market | (various) | (optional) |

| | | | |
|---|---|---|---|
| | participant identifier | | |
| D | Proprietary/Custom code | (various) | (optional) |

### 7.2.2 PartyIDSource for Broker of Credit:

| PartyIDSource (447) | | PartyID (448) | PartySubID (523) |
|---|---|---|---|
| B | BIC (Bank Identification Code) | <<BIC Code Value>> | (optional) |
| I | ISITC code for identifying directed brokers as per ETC Best Practices document (for use with PartyRole = Broker of Credit only) | <<ISITC-defined 3 character code>> | (optional) |
| D | Proprietary/Custom code | (various) | (optional) |

### 7.2.3 PartyIDSource for Settlement Location

| PartyIDSource (447) | | PartyID (448) | PartySubID (523) |
|---|---|---|---|
| B | BIC (Bank Identification Code) | <<BIC Code Value>> | (optional) |
| C | Generally accepted market participant identifier | CED = CEDEL<br>DTC = Depository Trust Company<br>EUR = Euroclear<br>FED = Federal Book Entry<br>HIC = Held In Custody<br>ICSD = International Central Securities Depository<br>NCSD = National Central Securities Depository<br>PNY= Physical<br>PTC= Participant Trust Company | (optional) |
| E | ISO Country Code *[for Local Market Settlement]* | << ISO Country Code Value >> | (optional) |

### 7.2.4 PartyIdSource for Buyer/Seller, Custodian, Intermediary or Agent

| PartyIDSource (447) | | PartyID (448) | PartySubID (523) |
|---|---|---|---|
| B | BIC (Bank Identification Code) | <<BIC Code Value>> | (optional) |
| H | CSD participant/member code (e.g. Euroclear, DTC, CREST or Kassenverein number) | <<CSD participant or member code>> | (optional) |

## *7.3 PartySubID*

The PartySubID is used to provide more detail about the PartyID. The PartyID field itself contains the actual value, and a repeating group of PartySubID and PartySubIDType fields may be optionally used to provide additional detail about the

party. For example, the PartySubIDType field can be used to identify the type of PartySubID value (i.e. "Firm", "Phone number", "Contact name", "Full legal name of firm", etc.)

Note that although there is a PartySubIDType, there is no “Role” field for a SubID. Example values are shown below:

| Value | PartySubIDType ( tags 803/805/786) |
|---|---|
| 1 | Firm |
| 2 | Person |
| 3 | System |
| 4 | Application |
| 5 | Full legal name of firm |
| 6 | Postal address (inclusive of street address, location, and postal code) |
| 7 | Phone number |
| 8 | Email address |
| 9 | Contact name |
| 10 | Securities account number (for settlement instructions) |
| 11 | Registration number (for settlement instructions and confirmations) |
| 12 | Registered address (for confirmation purposes) |
| 13 | Regulatory status (for confirmation purposes) |
| 14 | Registration name (for settlement instructions) |
| 15 | Cash account number (for settlement instructions) |
| 16 | BIC code |
| 17 | CSD participant/member code (e.g. Euroclear, DTC, CREST or Kassenverein number) |
| 18 | Registered address |
| 19 | Fund/account name |
| 20 | Telex number |
| 21 | Fax number |
| 22 | Securities account name |
| 23 | Cash account name |
| 24 | Department |
| 25 | Location / Desk |
| 26 | Position Account Type |

# 8. SETTLEMENT INSTRUCTIONS

## *8.1 Use of <SettlInstructions> Component Block*

The SettlInstructions component block is used to transmit settlement instruction details on an Allocation Instruction, Allocation Report, Confirmation or Settlement Instruction message.

- When used on an Allocation Instruction, Allocation Report or Confirmation message, this represents the settlement instructions that apply to a particular trade or order.
- When used on a Settlement Instruction message, this represents either standing instructions (to be used on future trades) or the instructions for a specific order (this usage is intended for the retail CIV market).

This component block can be used either to contain full settlement instruction details (i.e. settlement agent identities and account numbers) or a reference to a standing instruction database.

- When used to refer to instructions held on a standing instructions database, the StandInstDbType, StandInstDbName and StandInstDbID fields are used to specify the identify and name of the standing instructions database, and the identifier of the standing instruction record within that database. The NoDlvyInst repeating group should not be populated when using these fields.
- When used to specify settlement instruction details, the NoDlvyInst repeating group is used. Each member of that group holds one party's instructions for cash or securities settlement (o r both in the case of DVP). The SettlInstSource field identifies to whom the instructions belong, and the DlvyInstType field identifies whether the instructions are for securities or for cash.
- In both of these cases, the SettlDeliveryType field is used to identify the type of settlement LEIng represented by these settlement instructions, i.e. DVP (delivery vs payment), FOP (free of payment), hold in custody etc.

Where the component block is used to describe specific settlement instructions (i.e. using the NoDlvyInst repeating group), the number of entries in the NoDlvyInst repeating group is determined by the contents of the SettlDeliveryType field and the context of the message block (i.e. which message it is in). When used in an Allocation Instruction, Allocation Report or Settlement Instruction message, only the settlement instructions for the party generating the message need be specified. On a Confirmation message, both parties to the trade will have their settlement instructions specified. The matrix of usage of the NoDlvyInst repeating group is therefore as follows:

### 8.1.1 Allocation Instruction, Allocation Report or Settlement Instruction

| SettlDeliveryType | NoDlvyInst | SettlInstSource | DlvyInstType |
|---|---|---|---|
| 0 – Versus Payment | 1 | 1 (broker's), 2 (institution's) or 3 (investor's), depending on the identity of the originator of the message | S – securities |
| 1 – Free | 2 | 1 (broker's), 2 (institution's) or 3 (investor's), depending on the identity of the originator of the message | S – securities |

| | | | |
|---|---|---|---|
| | | 1 (broker's), 2 (institution's) or 3 (investor's), depending on the identity of the originator of the message | C – cash |

### 8.1.2 Confirmation

| SettlDeliveryType | NoDlvyInst | SettlInstSource | DlvyInstType |
|---|---|---|---|
| 0 – Versus Payment | 2 | 1 (broker's) | S – securities |
| | | 2 (institution's) | S – securities |
| 1 – Free | 4 | 1 (broker's) | S – securities |
| | | 1 (broker's) | C – cash |
| | | 2 (institution's) | S – securities |
| | | 2 (institution's) | C – cash |

The actual instructions themselves are held within the SettlParties component block inside the NoDlvyInst repeating group. This contains a repeating group of party identifiers and sub ids and is used to hold the identifiers of all parties involved in settlement (e.g. agent, custodian, depository) together with any required account numbers, registration details or similar.

## *8.2 SettlParty*

FIX supports the concept of a “SettlParty”, this being an organisation or individual connected in some way to the settlement of a financial transaction. Every SettlParty has a role (defining “what” the SettlParty is or what role it’s playing in the settlement process), an identifier, SettlPartyID (with a SettlPartyIDSource to identify the type of SettlPartyID) and any number of sub-identifiers (SettlPartySubID), each with a SettlPartySubIDType to define the type of sub-identifier.

### 8.2.1 SettlPartySubID

For the purposes of settlement instruction definition, the settlement party sub-identifiers can be taken to represent one of three things:

1. An alternative identifier for the SettlParty. For example, if the SettlParty’s primary identifier is its BIC code (expressed through its SettlPartyID with SettlPartyIDSource = B for BIC) then any other identifiers for the SettlParty (e.g. CSD participant number) can be expressed using a SettlPartySubID. For every SettlPartyIDSource that is commonly used to identify a SettlParty for settlement purposes, there is an equivalent SettlPartySubIDType.
2. An identifier of an account held at the SettlParty. Note that the convention is to hold the account details under the SettlParty at which the account is held, rather than under the SettlParty on whose behalf the account is held. For example, the account number of a custodian at an agent is held as a SettlPartySubID under the SettlParty representing the agent, not the custodian.
3. Additional information relating to the SettlParty, e.g. its full name, address, contact name, phone number etc.

When using the FIX SettlInstructions component block, it may be appropriate to provide a number of identifiers for the same SettlParty (e.g. both the BIC code and

CREST id for a CREST member agent bank). Only one of these can be held as a SettlPartyID – the other(s) must be held as SettlPartySubID(s). It does not matter which is held where.

# 9. MAPPING FIX to ISO15022

## *9.1 Towards the Rosetta Stone*

It is important to note that the ISO15022 standard has a consistent set of codes for what in FIX terms would be called the SettlPartyIDSource (or SettlPartySubIDType for sub-identifiers). Examples include:

- C – Country code
- P – Qualifier (BIC/BEI)
- R – Data Source Scheme/Proprietary Code
- Q – Name and address
- S – Alternate ID

In the interests of assuring STP, FIX fields and messages only map to ISO15022 options C, P, R or S (with a strong preference for P - BIC wherever possible) and SMPG guidelines will be followed throughout. There is no equivalent of 'Q' in FIX at the SettlParty level, though this is supported at SettlPartySubID level.
The ISO 15022 standard uses a similar methodology to the component blocks in FIX. This means that the same ISO tag can be used many times in the same message but its meaning depends on which message 'sequence' it is in. This is equivalent to the FIX concept of SettlPartyRole.

## *9.2 Some Worked Examples*

### 9.2.1 PSET

For example, a PSET BIC should be represented in FIX using these tags:

| FIX Tag | Value |
|---|---|
| 782 SettlPartyID | CEDELULL |
| 783 SettlPartyIDSource | B |
| 784 SettlPartyIDRole | 10 |

The mapping to a SWIFT tag would work here as follows:

1. FIX tag 782 is a SettlPartyID and therefore maps to SWIFT tag 95 (Party)
2. FIX tag 783 shows that the SettlPartyIDSource is a BIC and therefore maps to SWIFT option P.

We can now derive the correct SWIFT tag as 95P, which takes the format

**:Tag::Qualifier//BIC,** or in SWIFT shorthand **::4!c//4!a2!a2!c[3!c]** (where [3!c] represents the optional XXX characters at the end of an 8-character BIC).

So, concatenating the values within, or implied by, the FIX tags the mapping is:

**782 & 783::& 784 & //& 782** , or in the final message, **:95P::PSET//CEDELULL**

### 9.2.2 CSD Identifiers

ISO15022 allows a CSD identifier to be specified along with the type of identifier being used. For example:

> **:95R::DEAG/CRST/636** - Tag(Option):: (Qualifier)/(Data Source Scheme)/(Proprietary Code)

Here, the various tags have the following meanings:

- 95 (Tag) = PARTY
- R (Option) = The party will be identified by a data source scheme/ proprietary code
- DEAG (Qualifier) = Deliverer's agent
- CRST (Data Source Scheme) = Crest
- 636 (Proprietary Code) = participant ID at Crest.

In order to avoid having the full set of CSD identifier types listed as separate enumerations of PartyIDSource/PartySubIDType, FIX treats all such identifiers simply as CSD participant/member codes (PartyIDSource = H, PartySubIDType = 17). The type of participant/member code (e.g. Euroclear vs. DTC vs. CREST etc.) can be derived simply by looking at the instruction's settlement location (PartyRole = 10– equivalent to ISO15022 PSET). This is illustrated in the example below.

### 9.2.3 Putting it all together

**Settlement instructions for German domestic settlement with Citibank Frankfurt as local agent, into account 11921500:**

| <SettlParties> | | | | | |
|---|---|---|---|---|---|
| *Tag* | *Field Name* | | | *Value* | *Comments* |
| 781 | NoSettlPartyIDs | | | 3 | |
| → | *782* | *SettlPartyID* | | DAKVDEFF | PSET for German domestic settlement |
| → | *783* | *SettlPartyIDSource* | | B | BIC is used as the identifier in 782 |
| → | *784* | *SettlPartyRole* | | 10 | Settlement location (PSET) |
| → | *782* | *SettlPartyID* | | 7671 | Broker's agent's Kassenverein number |
| → | *783* | *SettlPartyIDSource* | | H | CSD participant/member code (e.g. Euroclear, DTC, CREST or Kassenverein number)<br>As the settlement location here is 'German domestic', this identifier is therefore a Kassenverein number |
| → | *784* | *SettlPartyRole* | | 30 | Agent – maps to SWIFT DEAG or REAG (depending on Side) |
| → | *801* | *NoSettlPartySubIDs* | | 1 | |
| → | → | *785* | *SettlPartySubID* | CITIDEFF | This agent's BIC code<br>This is held here as a PartySubID, though could also have been held as the PartyID with the Kassenverein number held as PartySubID instead |
| → | → | *786* | *SettlPartySubIDType* | 16 | BIC code |
| → | *782* | *SettlPartyID* | | 9427 | Broker or broker's custodian's Kassenverein number |
| → | *783* | *SettlPartyIDSource* | | H | CSD participant/member code (e.g. Euroclear, DTC, CREST or Kassenverein number) (KV no. in this case)<br>As the settlement location here is 'German domestic', this identifier is therefore a Kassenverein number |

| | | | | | |
|---|---|---|---|---|---|
| → | *784* | *SettlPartyRole* | | 27 (client) or 28 (custodian) | Deliverer/receiver of securities (or custodian) – maps to SWIFT DECU or RECU (depending on Side) |
| → | *801* | *NoSettlPartySubIDs* | | 1 | |
| → | → | *785* | *SettlPartySubID* | 11921500 | Securities account number |
| → | → | *786* | *SettlPartySubIDType* | 10 | Securities Account – maps to ISO15022 Tag 97 SAFE (Safekeeping account) |
| </SettlParties> | | | | | |

**SWIFT settlement instruction for an example trade, using settlement instructions derived from the above FIX data:**

| | |
|---|---|
| :16R:GENL<br><br>:20C::SEME//011204000064001<br>:23G:NEWM<br>:16S:GENL | |
| :16R:TRADDET<br>:94B::TRAD//EXCH/XETR<br>:98A::SETT//20011206<br>:98A::TRAD//20011204<br>:35B:ISIN DE0005557508<br>:16S:TRADDET | |
| :16R:FIAC<br>:36B::SETT//UNIT/3000,<br>:97A::SAFE//11921500<br>:16S:FIAC | Securities account number (taken from third SettlParty in the table above). |
| :16R:SETDET | |
| :22F::SETR//TRAD | |
| :16R:SETPRTY<br>:95R::DEAG/DAKV/7671<br>:16S:SETPRTY | Agent – the second SettlParty in the table above. The DAKV identifies the number 7671 as being a Kassenverein number and is derived from a combination of this party's SettlPartyIDSource (H - CSD code) and the SettlPartyID of the settlement agent. |
| :16R:SETPRTY<br>:95P:PSET//DAKVDEFF<br>:16S:SETPRTY | Settlement location – the first SettlParty in the table above. |
| :16R:SETPRTY<br>:95R::SELL/DAKV/9427<br>:16S:SETPRTY | Custodian/client – the third SettlParty in the table above. |
| :16R:AMT<br>:19A::SETT//EUR50700,<br>:16S:AMT | |
| :16S:SETDET | |

## 10. SSI Enrichment

The options for SSI population are:

1. No SSIs – parties to refer to internal or external databases
2. PSET only
3. Full enrichment

For option 3, some or all of the following fields (depending on market) can be considered as the minimum for full enrichment for DVP/RVP and free settlement:

- PSET
- Agent BIC
- CSD identifier (DTC 418, Crest 636, Euroclear 90895 etc)
- Securities account
- Cash account
- Investor id
- Registration number

# 11. SETTLEMENT INSTRUCTIONS MESSAGE

It may seem paradoxical to consider this message *after* the settlement-related contents of the various parties Component Blocks, but it is the contents and format of these blocks that determines the content and format of the settlement instruction message and the format of the SSI data stores to be held on the buy-side firms' infrastructure. The urgent task facing the industry is to tighten up the rules for transmitting and storing commonly-used settlement-related data fields *before* the message enters service.

## *11.1 Settlement Instructions*

The Settlement Instructions message provides the sell-side's, the buy-side's, or the intermediary's instructions for trade settlement. The SettlInstSource field indicates if the settlement instructions are the sell-side's, the buy-side's, or the intermediary's. This message has been designed so that it can be sent from the sell-side to the buy-side, from the buy-side to the sell-side, or from either to an independent "standing instructions" database or matching system or, for CIV, from an intermediary to a fund manager.
The Settlement Instructions message can be used in one of 3 modes (SettlInstMode):

1. To provide "standing instructions" for the settlement of trades occurring in the future. The message could either be sent in an 'unsolicited' fashion (i.e. a 'push'-style update from one firm to that firm's counterparties) or in response to a Settlement Instruction Request message. In either of these scenarios, this message can provide multiple settlement instructions.
2. To reject a Settlement Instruction Request message (e.g. unable to process request, no matching settlement instructions found).
3. To provide settlement instructions for a specific Order with a single account either as overriding or standing instructions to support matching. The ClOrdID field should be used to link the settlement instructions to the corresponding Order message.

The Settlement Instruction detail can be either explicitly specified (via the SettlInstructions component block) or can exist within an independent standing instructions database and can be referenced via the StandInstDbType, StandInstDbName, and StandInstDbID fields. See Volume 6 – Appendix 6-H for further details regarding the construction and formatting of settlement instruction details.

## Settlement Instructions

| *Tag* | *Field Name* | | *Req'd* | *Comments* |
|---|---|---|---|---|
| | *Standard Header* | | Y | MsgType = T |
| 777 | SettlInstMsgID | | Y | Unique identifier for this message |
| 791 | SettlInstReqID | | N | Only used when this message is used to respond to a settlement instruction request (to which this ID refers) |
| 160 | SettlInstMode | | Y | 1=Standing Instructions, ~~2=Specific Allocation Account Overriding, 3=Specific Allocation Account Standing ,~~4=Specific Order, 5=Reject SSI request |
| 792 | SettlInstReqRejCode | | N | Required for SettlInstMode = 5. Used to provide reason for rejecting a Settlement Instruction Request message. |
| 58 | Text | | N | Can be used to provide any additional rejection text where rejecting a Settlement Instruction Request message. |
| 354<br>355 | EncodedTextLen<br>EncodedText | | N<br>N | |
| 165 | SettlInstSource | | N | 1=Broker's Settlement Instructions, 2=Institution's Settlement Instructions , 3=Investor<br>Required except where SettlInstMode is 5=Reject SSI request |
| 11 | ClOrdID | | N | Required for SettlInstMode=4. |
| 60 | TransactTime | | Y | Date/time this message was generated |
| 778 | NoSettlInst | | N | Required except where SettlInstMode is 5=Reject SSI request |
| → | ***162*** | ***SettlInstID*** | N | Unique ID for this settlement instruction.<br>Required except where SettlInstMode is 5=Reject SSI request |
| → | ***163*** | ***SettlInstTransType*** | N | New, Replace, Cancel or Restate<br>Required except where SettlInstMode is 5=Reject SSI request |
| → | ***214*** | ***SettlInstRefID*** | N | Required where SettlInstTransType is Cancel or Replace |
| → | ***component block <Parties>*** | | N | Insert here the set of "Parties" (firm identification) fields defined in "COMMON COMPONENTS OF APPLICATION MESSAGES"<br>Used here for settlement location.<br>Also used for executing broker for CIV settlement instructions |
| → | ***54*** | ***Side*** | N | Can be used for SettlInstMode 1 if SSIs are being provided for a particular side. |
| → | ***460*** | ***Product*** | N | Can be used for SettlInstMode 1 if SSIs are being provided for a particular product. |
| → | ***167*** | ***SecurityType*** | N | Can be used for SettlInstMode 1 if SSIs are being provided for a particular security type (as alternative to CFICode). |
| → | ***461*** | ***CFICode*** | N | Can be used for SettlInstMode 1 if SSIs are being provided for a particular security type (as identified by CFI code). |
| → | ***168*** | ***EffectiveTime*** | N | Effective (start) date/time for this settlement instruction.<br>Required except where SettlInstMode is 5=Reject SSI request |
| → | ***126*** | ***ExpireTime*** | N | Termination date/time for this settlement instruction. |
| → | ***779*** | ***LastUpdateTime*** | N | Date/time this settlement instruction was last updated (or created if not updated since creation).<br>Required except where SettlInstMode is 5=Reject SSI request |
| → | ***Component block <SettlInstructions>*** | | N | Insert here the set of "SettlInstructions" fields defined in "COMMON COMPONENTS OF APPLICATION MESSAGES" |
| → | ***492*** | ***PaymentMethod*** | N | For use with CIV settlement instructions |
| → | ***476*** | ***PaymentRef*** | N | For use with CIV settlement instructions |
| → | ***488*** | ***CardHolderName*** | N | For use with CIV settlement instructions |
| → | ***489*** | ***CardNumber*** | N | For use with CIV settlement instructions |
| → | ***503*** | ***CardStartDate*** | N | For use with CIV settlement instructions |
| → | ***490*** | ***CardExpDate*** | N | For use with CIV settlement instructions |
| → | ***491*** | ***CardIssNum*** | N | For use with CIV settlement instructions |
| → | ***504*** | ***PaymentDate*** | N | For use with CIV settlement instructions |
| → | ***505*** | ***PaymentRemitterID*** | N | For use with CIV settlement instructions |
| | *Standard Trailer* | | Y | |

# 12. APPENDIX 1 – RELATIONAL INTEGRITY

**Notes on Relational Integrity and Compatibility with ISO15022**

The FIX 4.4 post-execution messages have been designed to be compatible with the ISO15022 standard. To ensure that all parties can translate a FIX message into a SWIFT message without ambiguity, it is essential that the information on Allocation Instructions and Confirmations conforms to certain relational integrity rules. This is particularly applicable to the data within the component blocks. The ability to use these blocks to define any number of parties gives considerable flexibility, but there are certain pitfalls. The same *SettlPartyIDRole* should never repeat within the same <SettlParties> block. For example, this slightly contrived combination would be allowed because even though the values in *SettlPartyID* and *SettlPartyIDSource* are identical, the values of tag 784 (784=30 and 783=27) are unique.

| <SettlParties> | | | | |
|---|---|---|---|---|
| *Tag* | *Field Name* | | *Value* | *Comments* |
| 781 | NoSettlPartyIDs | | 2 | |
| → | *782* | *SettlPartyID* | CITIGB21XXX | |
| → | *783* | *SettlPartyIDSource* | **B** | BIC |
| → | *784* | *SettlPartyRole* | **30** | Agent |
| → | *782* | *SettlPartyID* | CITIGB21XXX | |
| → | *783* | *SettlPartyIDSource* | **B** | BIC |
| → | *784* | *SettlPartyRole* | **27** | Buyer/Seller (receiver/deliverer) |
| </SettlParties> | | | | |

However, this equally contrived combination would not be allowed because the values in *SettlPartyRole* are identical (784= 4 and 784=4) even though the BICs are different.

| <SettlParties> | | | | |
|---|---|---|---|---|
| *Tag* | *Field Name* | | *Value* | *Comments* |
| 781 | NoSettlPartyIDs | | 2 | |
| → | *782* | *SettlPartyID* | DAKV1234 | |
| → | *783* | *SettlPartyIDSource* | **C** | Generally accepted market code |
| → | *784* | *SettlPartyRole* | **4** | Clearing firm |
| → | *782* | *SettlPartyID* | DEUTFF2LXXX | |
| → | *783* | *SettlPartyIDSource* | **B** | BIC |
| → | *784* | *SettlPartyRole* | **4** | Clearing firm |
| </SettlParties> | | | | |

# 13. APPENDIX 2 – IDENTIFYING FUNDS AND LEGAL ENTITIES

## *13.1 Statement of Problem*

Main areas of concern

1. Legal and compliance
2. Processing
3. Risk

In summary:

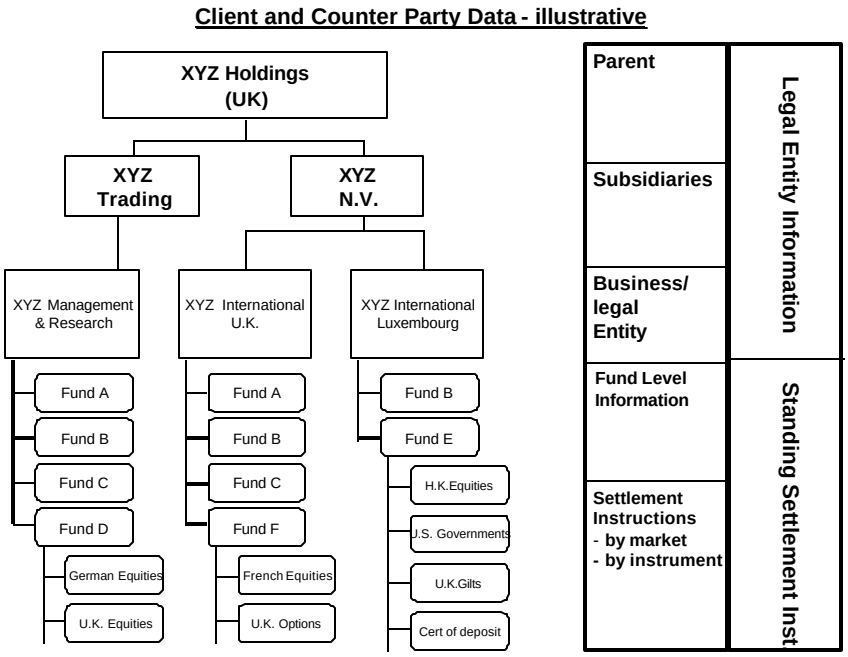
- No commonly agreed identifier for regulated entities.
- Difficult for counterparties to identify the other side's contracting entity.
- No commonly agreed identifier for funds managed by these entities.
- No existing method (outside VMUs) for real-time enrichment of SSIs on confirmation messages.
- Increased operational and credit risk.
- Difficult to group entities together for firm-wide risk exposure purposes.
- Failure to meet G30 requirements.

## *13.2 Assumptions*

It is assumed that:

1. A new identifier, to be called the **Legal Entity Identifier (LEI)** and based on the existing 8-character BIC code structure, will be created.
2. The LEI will **not** be the same as any existing BIC. It is simply based on the logic of the BIC and is an ISO15022-standard identifier.
3. The LEI will not be used for SWIFT messaging.
4. The LEI will be used to identify a unique regulated entity, or in the case of unregulated entities, the fund or organisation to which an order (or part of an order) is allocated.
5. An issuing authority for new identifiers will be appointed.
6. Agreement of all parties to the proposed structure will be obtained.
7. Data providers and regulators will agree to carry the new data fields.
8. The proposed hierarchy is designed to meet compliance (EDD, KYC, AML etc) and processing requirements only.

### 13.2.1 Example Data Structure




## *13.3 Scope*

While all elements of the structure are in scope, the most pressing need is to create those relationships which most directly affect STP:

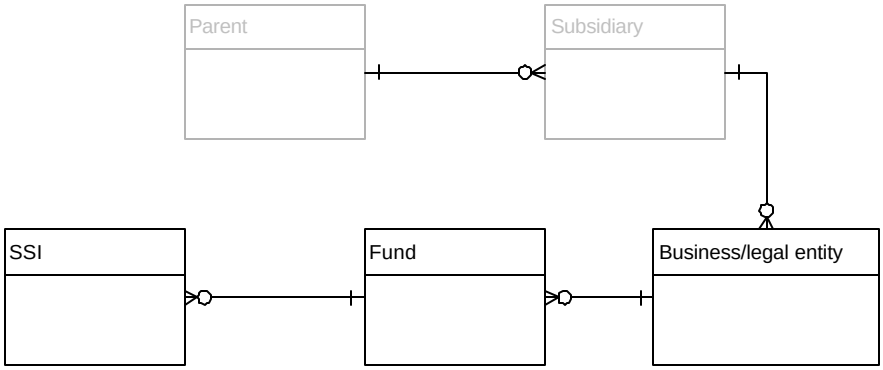
1. Legal/business entity
2. Fund
3. SSI.

The relationship between those entities above the Legal/business Entity in the data structure shown in section 3.2.1 will be mainly used for risk calculations and are thus out of scope for security processing. Individual firms' risk management departments can use existing data sources such as D&B, Bankers' Almanac etc to link their risk groupings to the Business Entity structure proposed by this paper. We have not attempted to impose a formal structure for those entities *above* the business/legal entities in the hierarchy because individual departments (risk, CRM, financial, compliance etc) will want to create their own groupings for this hierarchy.

## *13.4 Suggested Implementation*

### 13.4.1 Ideal Structure

The basic problem to be solved is to create this relational structure:




As explained earlier, the hierarchy above the legal/business entity level (Parent and Subsidiary) is more relevant to risk considerations and is therefore out of scope for trade processing and STP.

#### 13.4.1.1 Legal Entities

National regulators and exchanges hold data on their members and legal entities which they regulate, but each of these organisations present their data in different forms and it is therefore difficult to consolidate the available information.

The proposed solution is to work with SWIFT (possibly acting as issuing authority), national regulators and exchanges to ensure that each regulated business entity and exchange member firm is assigned a unique **8**-character LEI (based on the BIC code) and that the LEI is published on the exchanges' and regulators' databases as a mandatory field. For example:

| Trading entity |
| --- |
| LEI: CITIGB2L |

This will make it far easier to carry out know-your-client (KYC), Patriot Act and anti-money laundering (AML) compliance procedures. The LEI could also be used for regulatory reporting.

#### 13.4.1.2 Funds

One of the most difficult problems faced by the industry is that there is no globally agreed fund identifier. Part of this gap is filled by OMGEO's Alert product, but usage is far from universal.

Again, the structure of the BIC code supplies a possible solution for creating a LEI hierarchy that uniquely identifies funds.

The last 3 characters of the **11** character BIC code are intended to be used to identify branches of the parent institution represented by the 8-character BIC. However, given that these 3 characters can be arranged in a very large number of unique permutations they could also be used to identify individual funds without conflicting with their intended use as branch identifiers.

The number (P) of 3 -letter groups (r) that can be created from 26 letters (n) can be calculated using the following formula:

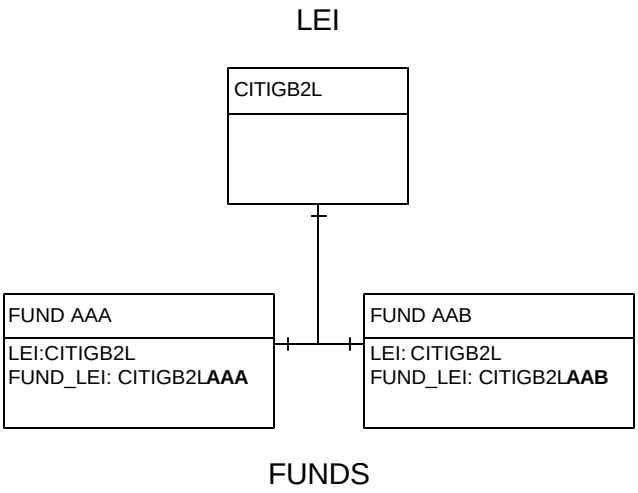
$$P(n,r) = \frac{n!}{(n-r)!}$$

or in this example $P(n,r) = \frac{26!}{(26-3)!}$ = **15,600** 3-letter groups.

(If, in addition to using the letters A to Z, the numbers 0 to 9 are included, the available number of unique groups goes up to 42,840). If characters are allowed to repeat within the groups, AAA, AAB, for example, the number of groups is even higher ($36^3$=46,656)

Therefore, we can create a unique LEI composed of the first 8 characters of the BIC to represent the legal entity *plus* a 3-letter group to give a possibility of at least 15600 (42,840) fund ids. We can call this the ***Fund LEI***.

The next part of the relational structure is thus created:




There is no conflict between this new field and existing identifiers such as the Alert acronym/access code pairing. For example, an existing Alert fund under Omgeo Acronym BIGBANK1 with Omgeo Access Code 123 might have a Fund LEI of BIGBNY21XYZ, and Alert would show both values.

It is also important to note that there would be no attempt to standardise fund identifiers across different fund managers. For example, if the Teacher's Pension Fund of Texas was managed by 2 fund managers, each would create their own distinct 3-character Fund LEI. Similarly, when a fund's trustees remove the mandate from one fund-manager and give it to another, the receiving manager will assign the fund an new Fund LEI.

#### 13.4.1.3 Standing Settlement Instructions (SSIs)

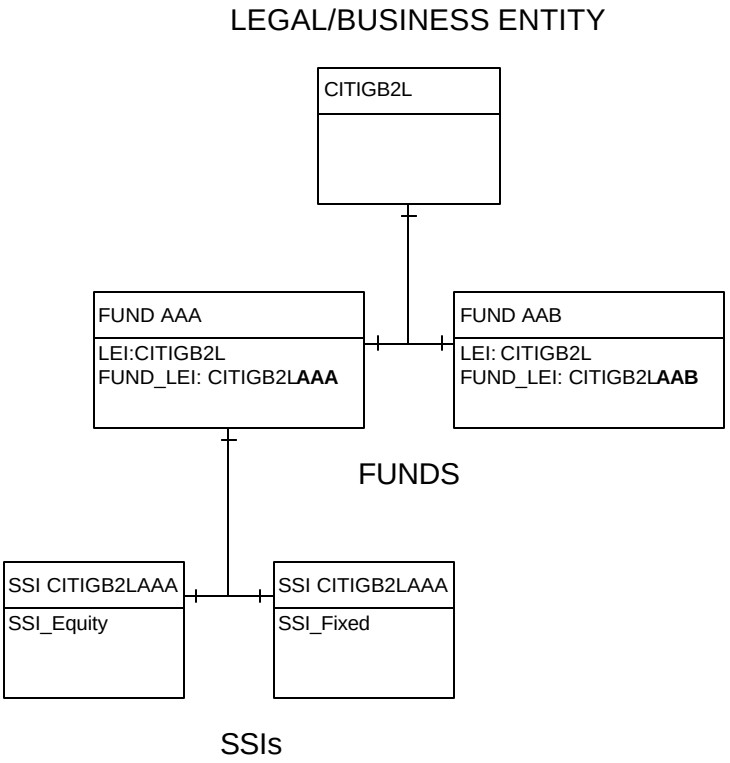
Assuming that:

1. Creation of a globally-accepted, industry-wide SSI database is not feasible.
2. all organisations which publish data about funds or broker entities use the Fund LEI as the unique identifier.
3. all data vendors agree to take this additional field.

4. SSIs will conform to the ISO 15022 standard

then the available solutions for transmitting this data are:

1. Where there is no data vendor, brokers and buy-side firms can exchange this information bilaterally and save it in their own databases.
2. Participants can use "just-in-time" enrichment of allocation and confirmation messages with SSI information.




## *13.5 Just-in-Time SSI Enrichment*

If all parties to a trade can agree on how to identify themselves to each other and the buy-side use the proposed Fund LEI model to identify their funds, there is no longer any need for the industry to use and maintain costly shared account and SSI databases. Instead, custodians would send fund managers a periodic update file (in ISO15022-compliant format) of basic SSI details which would be captured in a database within the buy-side's OMS. This would have the advantage of keeping the responsibility for SSI maintenance with the custodians but allow fund managers to enrich their allocation messages with up-to-date SSIs. With their smaller number of SSIs, broker-dealers would maintain their own SSI data with which they would enrich their confirmations. Both sides would match the other's SSIs for settlement channel compatibility, and if the match is successful, would then instruct for settlement. There would also be a major gain for sell-side firms who would no longer have the maintenance overhead of maintaining multiple SSIs for each client fund. The key to the success of this flow will be strict adherence to agreed data standards.

## 13.6 Regulatory Considerations

One of the keys to the success of this approach is its acceptance by regulators and exchanges. It is likely that these bodies will welcome a global standard for identifying their member entities that does not conflict with existing identifiers, although there may be a reluctance to make the technology investment required to prepare their databases for the use of LEIs. Also, the adoption of the LEI structure directly addresses and number of the issues and action plans raised by the recently published G30 report, 'Global Clearing and Settlement, a Plan of Action.'

## 13.7 Administration and Control

The issue of new LEIs and Fund LEIs will need to be controlled centrally by a single registration and issuing agency such as SWIFT. The service will need to be available H24, and the buy-side must be able to request and receive new Fund LEIs within 30 minutes. All requests for new LEIs at the entity level will be subject to compliance and legal checking in accordance with the governing law of their regulator. Although the creation of new entities will be a relatively rare event, there will be an administrative overhead involved: one option would be for the new entity firstly to request registration with a national regulator and for the regulator to make the request to the issuing agency for a new LEI. It is assumed that buy-side firms would not be willing to pay the full initial set-up costs involved in the creation of LEIs and Fund LEIs and that sell-side firms would be expected to foot some of the bill. This issue will affect any new identifiers that the industry chooses to adopt.

## 13.8 Risk

### 13.8.1 Project Risk

A number of issues and risks face this proposal:

- ROI proves insufficiently compelling to persuade all parties to adopt the solution.
- Buy-side OMS vendors unwilling to invest in the necessary changes to their systems.
- Regulators do not adopt the new identifiers.
- Control and admin overheads prove too high.
- Adherence to industry data standards is poor.
- Industry participants unwilling to face the start-up and maintenance costs.

### 13.8.2 Operational Risk Reduction

One of the major potential advantages of the approach outlined in this paper is the reduction of operational risk. All participants' cost per transaction will reduce as will the number of unmatched and failed trades. With the advent of Basel 2, any decrease in operational risk should be seen as a strong selling point for the LEI model.