Challenges in Adopting a Common Domain Model for Securities Finance

Regulations such as SFTR are driving discussion in the securities finance industry around how to agree on standardised processes and workflows. One idea is the Common Domain Model (CDM), similar to the initiative by ISDA in the derivatives space. Is there a strong argument in favour of a standardised domain model, and what challenges need to be overcome to get there?

BY MARTIN WALKER, VP, PRODUCT MANAGEMENT, SECURITIES FINANCE
AND COLLATERAL MANAGEMENT, BROADRIDGE

Since 2016, the International Swaps and Derivatives Association (ISDA) has worked to create a Common Domain Model that attempts to standardise the data representation of derivatives trades and trade events. The idea is to present a unified data format that all market participants can agree on, centrally managed by ISDA. A range of vendors and market participants are working to adopt CDM for specific opportunities in OTC derivatives. If successful, this new standardisation effort could reduce costs and drive efficiencies across the industry.

In securities finance, the upcoming Securities Finance Transactions Regulation (SFTR) is a catalyst for reconsidering standardisation models. Fear of SFTR is making people wonder whether a CDM will ultimately provide a roadmap for the work that is needed to help deal with the complexity of SFTR.
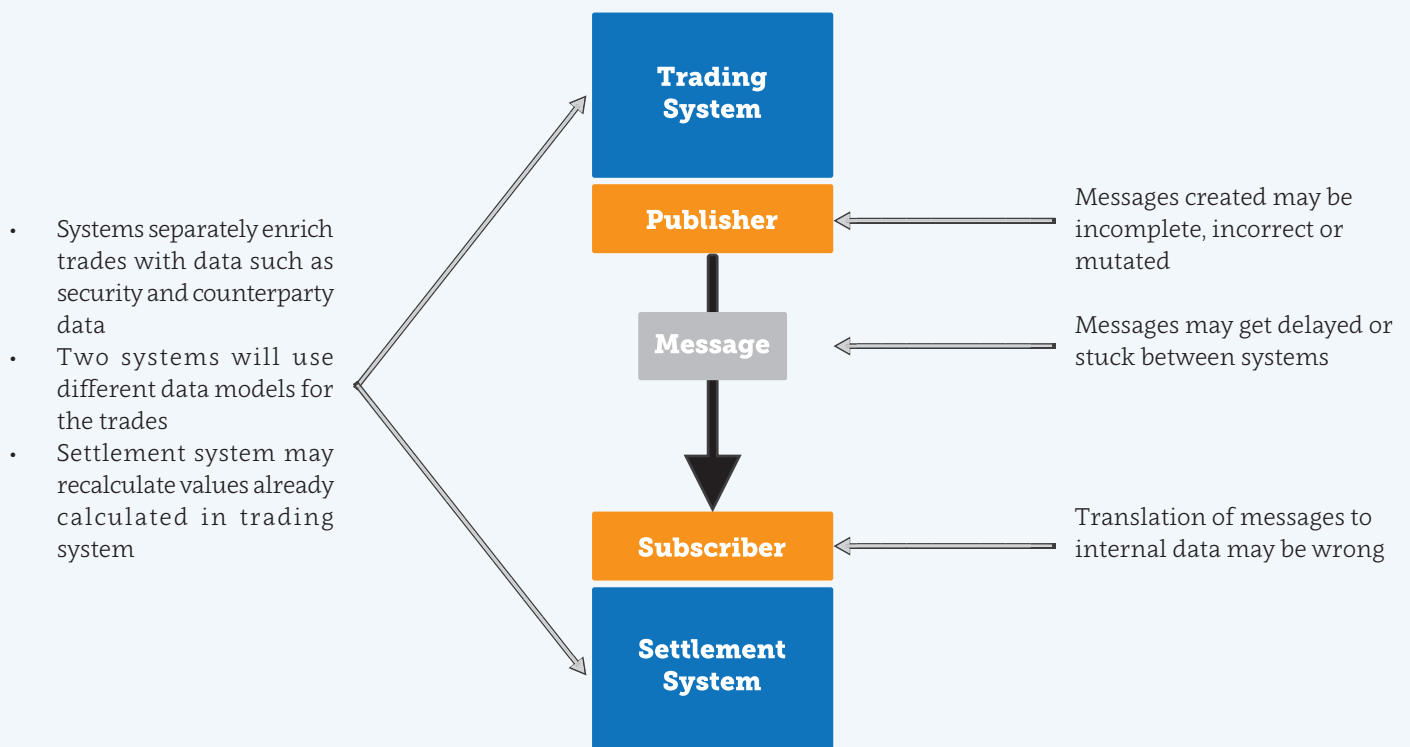
Past attempts at standardisation

Challenging times in capital markets mean the search continues for an effective way to reduce the costs of the infrastructure and support functions behind trading desks. This is not new, and, as I documented in my book, *Front-to-Back: Designing and Changing Trade Processing Infrastructure[1]*, there have been cycles of transformation based on the same ideas. These include "Simplification", "Front-to-Back Re-Engineering", "Front-to-Back systems" and "Functionalisation" (which is the complete opposite of a front-to-back system). A great deal of money has also been spent creating systems designed to sit between the front and back office in order to clean up trade data and provide a consolidated view of it.

For more than two decades, there's been a transition from monolithic systems, that had little interaction with other systems (internal or external), to today's situation of core systems sitting in a web of connections to internal and external systems. The need to communicate data between systems drove adoption of various formats for exchanging data about trades in the form of machine-readable messages. Even if the data models used in systems were not the same, standardising message formats meant different systems could communicate. However, they had to have standard interfaces for generating and consuming messages. Large scale adoption of standardised messaging in financial services took off in the 1970s with the creation of the SWIFT network for payments.

Despite a desire for standardisation, it took until 1992 before there was significant movement towards harmonising trade messages. This was the creation of the Financial Information eXchange (FIX)

Exhibit 1: Sources of breaks

- Systems separately enrich trades with data such as security and counterparty data
- Two systems will use different data models for the trades
- Settlement system may recalculate values already calculated in trading system

Trading System

Publisher — Messages created may be incomplete, incorrect or mutated

Message — Messages may get delayed or stuck between systems

Subscriber — Translation of messages to internal data may be wrong

Settlement System

Source: *"Front-to-Back: Designing and Changing Trade Processing Infrastructure" by Martin Walker, published by Risk Books*

FEATURE

protocol. FIX was initially developed to support the exchange of equities trade data between Fidelity Investments and Salomon Brothers and its adoption grew steadily as market participants saw its benefits. FIX had a widespread impact. However, the drive towards standardising the messaging for more complex products such as derivatives and securities finance trades began in 1999 with the creation of Financial Products Mark-Up Language (FpML). FpML has achieved fairly wide-scale adoption for both internal and external communication and gained the sponsorship of ISDA. However, there are limits to its usefulness.

Unlike SWIFT, there is no single network provider that can manage the transport of messages and impose standards. Trading also sees a greater rate of change in terms of new products being created than payments. New products are constantly being launched and old products modified. The rate of change, combined with a limited scope to impose standards has meant that FpML has mutated into many forms, particularly for internal use.

The degree of success of all these approaches has been, to put it politely, highly variable. An uncomfortable number of projects have ended in spectacularly expensive failure. Does anyone think in retrospect that ideas such as feeding all of a firm's poor-quality trade data into a "Big Data" solution, in the hope that emerging technologies such as artificial intelligence, blockchain or some future technology would eventually fix the problem was a good idea? In large part, failures resulted from attempting to satisfy a general desire for better infrastructure with a "miracle technology" without considering which specific problems should be solved and how well the solution matched the problem.

Infrastructure complexity

Examining the typical infrastructure of capital markets firms shows multiple systems recording and processing the same trades. In the simplest example, there is a front office trading system and back office settlement system. Both will typically store records of trades but in different formats i.e.

they have different data models. Different data models create the risk that two systems interpret the same economic transaction in different ways. This, in turn leads to "breaks" when the systems are reconciled. Differences in data models and business logic are a key cause (see Exhibit 1).

Generally, there are many more internal systems recording and processing the same set of trades. All of this internal complexity is mirrored by infrastructure of the firm's counterparties. Third parties such as agents, brokers, CCPs, trading platforms and other market infrastructures add to the complexity by storing their own representation of trades.

Building a data model

Recording a real-world activity, particularly something as complex as a securities finance trade, requires a structure: this structure consists of a set of attributes, such as start date, security identifier and trade type.

For each of those attributes it will typically have a label e.g. "TradeType". Depending on the type of system, it could also have a

Exhibit 2: Trade models for data elements
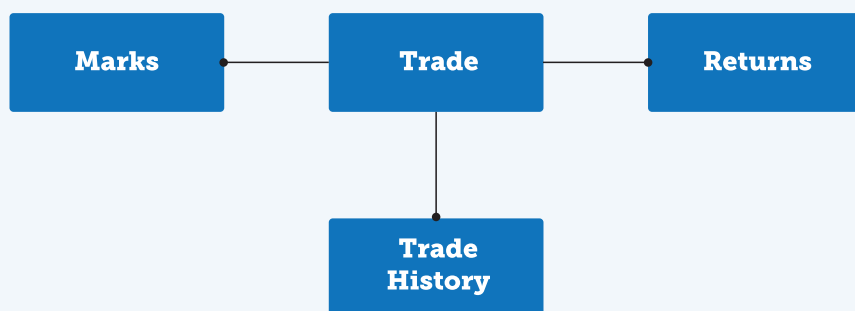
One trade may have many returns, multiple marks and a series of changes or corrections to the core details. These are one-to-many relationships.

specified type of attribute such as text or numeric. In addition, relationships need to be modelled to other types of data such as counterparties or links between different elements of a trade.

For trade types whose characteristics change over time (such as stock lending trades), data models must reflect those changes. For instance, a securities lending trade may have a series of returns of stock, a series of "Marks" (fee and/or cash) and series of changes to the core trade information. There are many ways to model the relationships between the elements of a trade (see Exhibit 2).

For some purposes, such as calculating the balance sheet impact of a trade, only the current view of a trade is required. For others, such as billing, the history of the trade over the billing period is required.

Practical implications of a CDM

What does it really mean to adopt a Common Domain Model? There are already standard (but widely abused) formats for messages. If a CDM is to emerge that has an impact beyond messaging, it requires fundamental changes to the design of systems. There are several possible models for adoption.

• Building systems that work off a shared centralised data store. This takes us to the world of "micro-services". Some banks have infrastructure designed this way in low latency trading, but it can be expensive and hard to maintain.

• Building systems that work off data stored in a distributed ledger using "Smart Contracts". This still requires considerable work, and replacing existing systems with immature new technologies can be high risk as well as expensive.

• Building systems that use shared or inhouse software working off data that is kept harmonised in both structure and content by continuous real-time matching and machine-readable "Product Definition Agreements," as described in the article, *Bridging the Gap between Investment Banking and Distributed Ledgers.*[2]

A Solution in Search of a Problem?

Given the potentially enormous costs associated with replacing or substantially rewriting infrastructure, the question has to be asked: is CDM a solution in search of a problem? It is quite likely a CDM for derivatives or securities finance will be discussed at length but have no real impact. The costs of adoption could easily exceed the benefits by several multiples. There is only one thing harder than persuading people in capital markets to agree standards: getting them to continue conforming to them in the absence of someone with a very large stick.

So, should the industry give up on the idea? That may be premature in the pre-SFTR world. It is likely that once regulators start analysing the reported data in detail, an early reaction could be: "Why are there so many trades that match on Unique Transaction Identifiers (UTIs) and Legal Entity Identifiers (LEIs) but break on core economic fields?" The answer may be that it is possible in many business segments for counterparties to book trades in significantly different ways but still manage to (mostly) settle trades correctly and agree billing. History and logic suggest that standardisation in capital markets can have large benefits. However, the best reason for pragmatically looking at a CDM for securities finance is to allow the industry to shape the standardisation agenda before standardisation is forced on the industry by regulators.

References:

1. "Front-to-Back: Designing and Changing Trade Processing Infrastructure," Martin Walker, 2018, available at https://www.amazon.co.uk/Front-Back-Designing-Processing-Infrastructure-ebook/dp/B07KFM9KYD

2. Bridging the gap between investment banking and distributed ledgers," The Journal of Financial Transformation", November 27, 2019, available at https://www.capco.com/Capco-Institute/Journal-46-Automation/Bridging-the-gap-between-investment-banking-infrastructure-and-distributed-ledgers

Martin Walker is the Head of Product Management for Broadridge's Securities Finance and Collateral Management division. Former roles include Global Head of Securities Finance and Treasury IT at Dresdner Kleinwort and Global Head of Prime Brokerage Technology at RBS Markets. He has also worked as a consultant and researcher in capital markets with several papers published. He contributed to the new book "Evidence-Based Management – How to Use Evidence to Make Better Organizational Decisions" and his first book on Capital Markets infrastructure will be published later in the year. He has an MSc in Computing Science from Imperial College, London and a BSc in Economics from the London School of Economics. He is also a fellow of the Center for Evidence-based Management.

Global Securities Finance and Collateral Management Solutions

- Optimize funding and collateral decisions
- Reduce counterparty and operational risks
- Enable efficient and high-growth operations
- Meet regulatory and market requirements

Technology
Communications
Data and Analytics

Broadridge.com/cm

Broadridge®