

**Business & Technical Overview
Common Domain Model (CDM™) and
Digital Regulatory Reporting (DRR)**



Antitrust Statement

- We would like to remind all participants for this training session on CDM and Digital Regulatory Reporting that they act independently in the market. Each market participant must use its own independent judgment with respect to its own business operations.
- Participants are reminded that no information regarding trading activities or business strategy should be exchanged or discussed.
- The purpose of the training is to focus on the Common Domain Model and Digital Regulatory Reporting. The conversation should remain focused on the meeting agenda presented for this training.
- By participating in this training, you agree not to use this process to accomplish any anticompetitive goal, and you agree that you will not use any information obtained during this training for any anticompetitive purpose.
- Please note that the contents of this training are confidential.

Morning Agenda: Business Overview

1) Setting the Scene: 09:00 – 09:15

2) Business overview: CDM & DRR- what, why and value: 09:15 – 10:15

- Background and history
- How CDM and DRR work together to fulfil the need for digitised reporting
- Business drivers and benefits
- High-level overview of the PoC Approach
- Strate Contribution

3) Adoption Models and Use Cases: 10:15 – 11:00

- How firms are adopting CDM and DRR in practice
- Benefits adopting firms have realised
- Considerations specific to South African firms

4) Networking and coffee break: 11:00 – 11:30

5) Business Q&A and discussion: 11:30 – 12:30

6) Lunch: 12:30 – 13:30

Afternoon Agenda: Technical Deep Dive

7) Building Transactions with the CDM: 13:30 – 14:45

- CDM core concepts from a technical perspective
- Construction of CDM transactions and lifecycle events
- Representation of products, events and core objects
- How CDM enables machine-executable logic

8) Coffee break: 14:45 – 15:15

9) From CDM to DRR- a Technical Walkthrough: 15:15 – 16:15

- How DRR consumes CDM-based transactions and events
- Mapping regulatory logic to CDM structures
- End-to-end walkthrough
- Examples
- State Technical Input

10) Technical Q&A and discussion: 16:15 – 16:45

11) Closing and Next Steps: 16:45 – 17:00

Setting the Scene



What you heard yesterday

SUMMARY - PART 2

Operations & Readiness

strateISDA

RDF

Day 1 Summary: Operations & Readiness

**Operational Timing**

T+1 10:00 UTC: Strict deadline for UTI bilateral dissemination.

Query Protocol: Never create competing UTIs; always trigger bilateral queries.

Strate Allegements: Leverage built-in TR pre-reporting trade copies.

**Trade Association Power**

Harmonised SOPs: Shared CFD UPI mapping and lifecycle event matrices.

Error reporting: Metric-based error notification materiality thresholds. **DRAFT**

Collective Voice: Constructive dialogue with FSCA and Strate.

**12-Month Execution**

Buy vs Build: Choose technology architecture early based on scale.

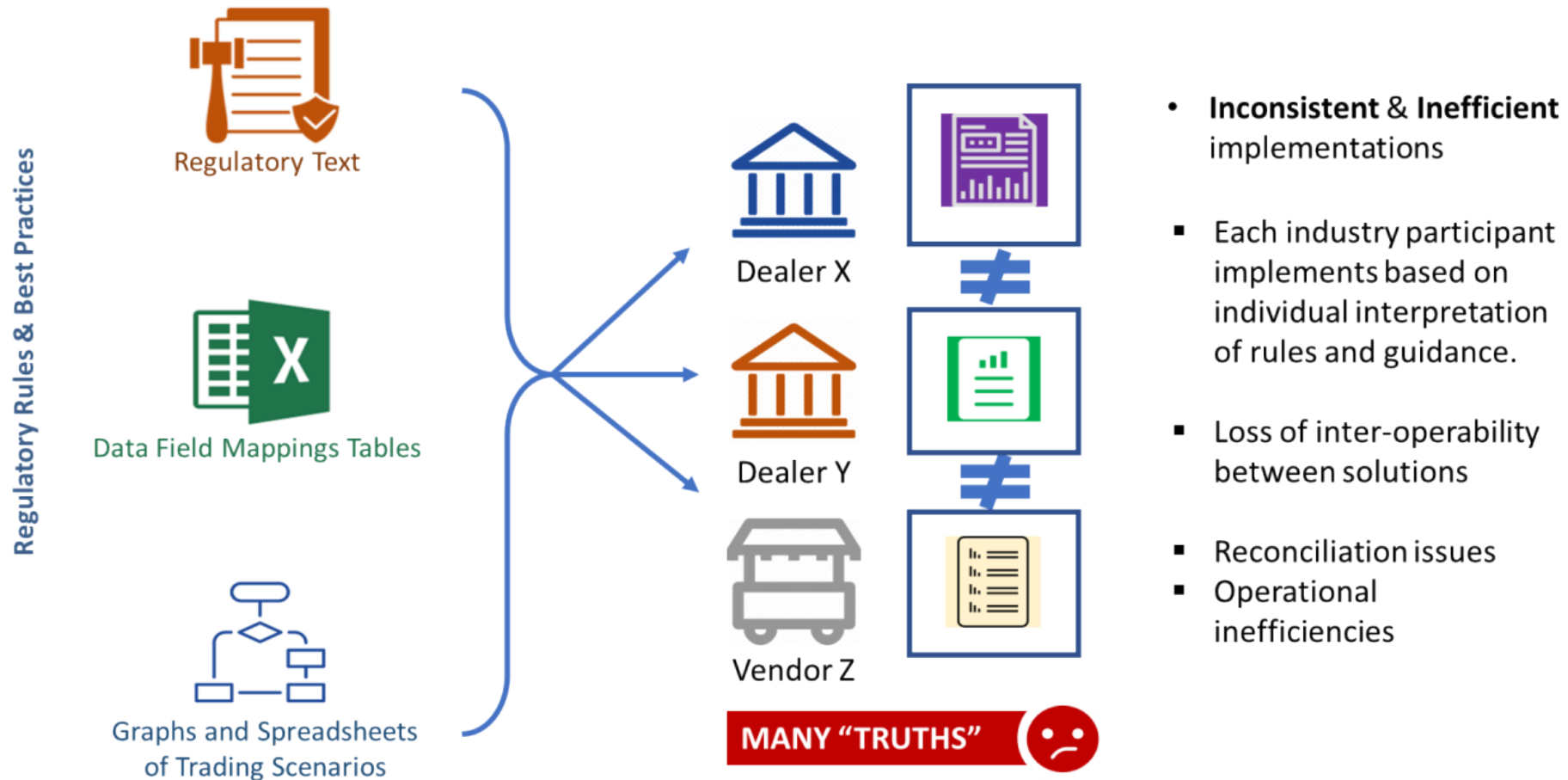
Backfilling: Validate open trade migrations in non-prod environments.

UAT vs PROD: How you will plan your testing, integration vs end to end, before Production release, will be key.

 Success requires tight alignment across legal, technology, trade booking, and compliance teams.

Regulatory Reporting Implementation: The Problem Statement

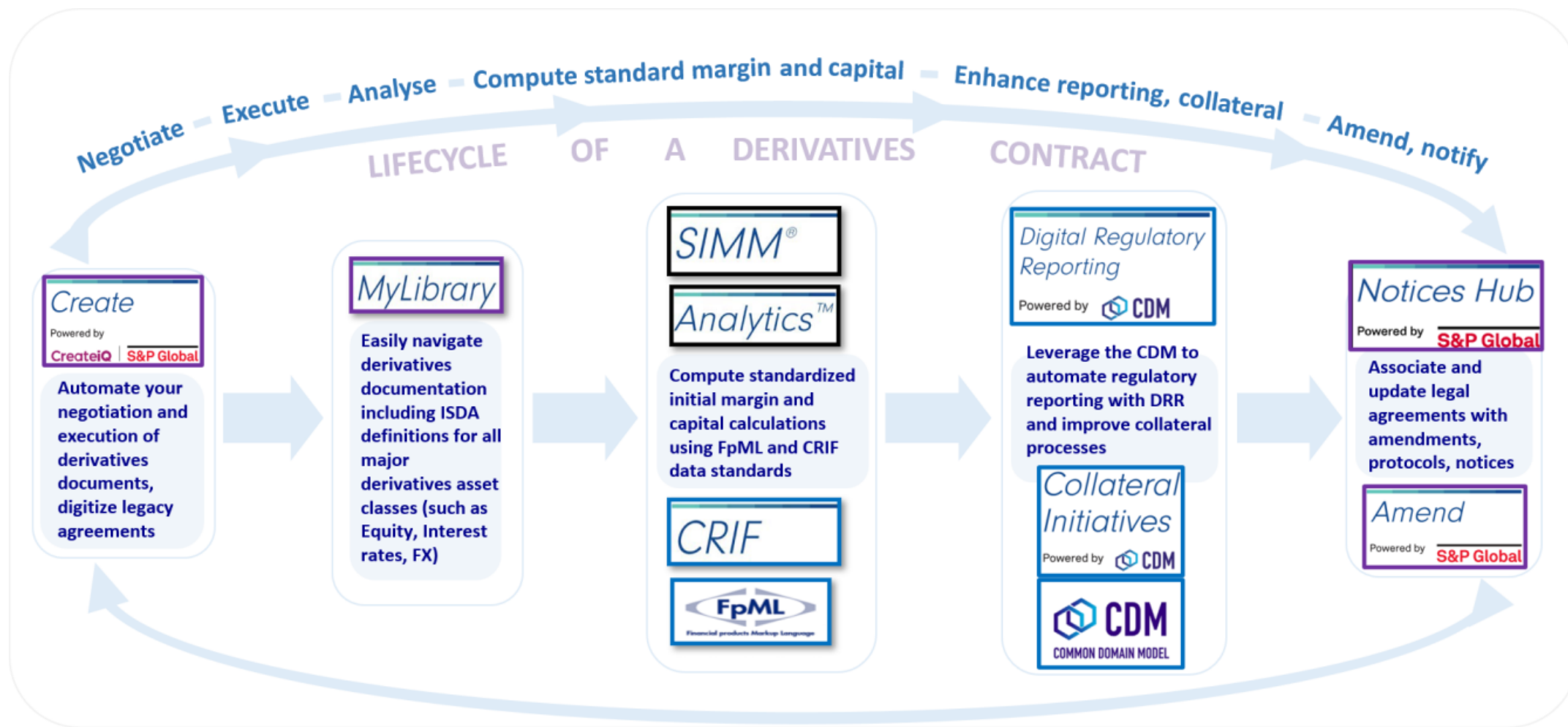
Industry firms currently build based on individual rule interpretations



There has to be a better way!

Navigating Financial Derivatives: ISDA's Digital Solutions

ISDA's suite of mutualized digital solutions cover **Legal**, **Data** and **Risk**. Each solution helps firms standardize, optimize and automate key aspects of the derivatives lifecycle processes cost-effectively and at scale.

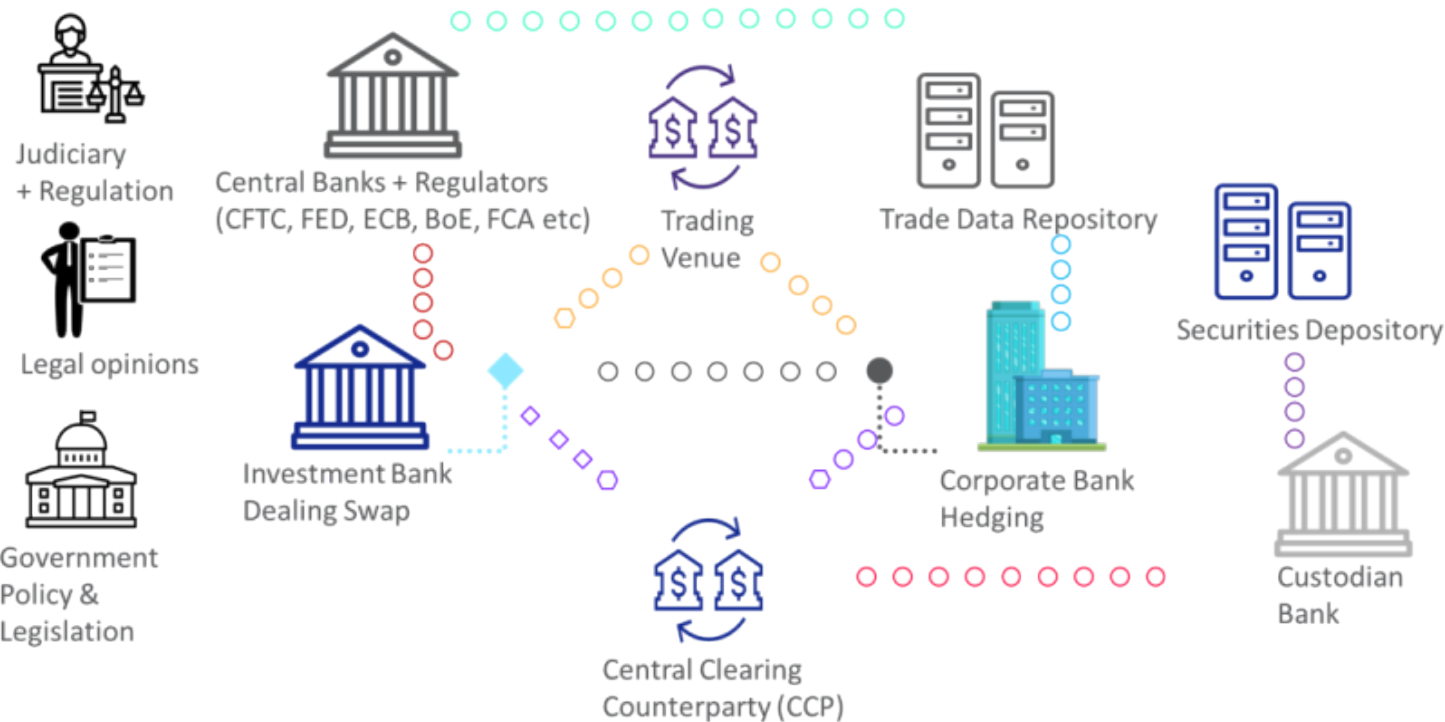


Business Overview: CDM DRR- what, why and value



Catalyst for Change- Market Structural Challenges (this isn't just Reg Reporting at fault)

All parties store trade data in different formats & make lifecycle changes to these records inconsistently



What is the true “truth” at any point in time?

Differences in booking models lead to real world events in those models producing different outcomes:

- Reconciliation breaks
- Valuation differences
- Collateral disputes
- Reporting mismatches
- Operational inefficiency
- Settlement failures
- Barriers to automation

What is the CDM?

The Common Domain Model (CDM) is a standardised, machine-readable and machine-executable blueprint for how financial products are traded and managed across the transaction lifecycle.

Dimensions of the CDM:

Product	Definitions of tradeable products qualified by their economic terms
Event	Data structures to represent the lifecycle events of financial transactions
Legal Agreement	Digital representation of the legal agreements that govern transactions
Process	Translates the technical standards that support those industry processes into a standardised machine-readable and machine-executable format
Reference Data	Reference data components that are specifically needed to model the other dimensions
Mapping	Mapped to a set of alternative data representations including FIX, FpML, ISO20022

The CDM is **NOT** an application in and of itself, but can be implemented within one
Composability allows for re-use of components for efficiency

Product Coverage

The scope of contractual products in the current model are summarized below:

- **Interest rate derivatives:**

- Interest Rate Swaps (incl. cross-currency swaps, non-deliverable swaps, basis swaps, swaps with non-regular periods, ...)
- Swaptions
- Caps/floors
- FRAs
- OTC Options on Bonds

- **Credit derivatives:**

- Credit Default Swaps (incl. baskets, tranche, swaps with mortgage and loans underliers, ...)
- Options on Credit Default Swaps

- **Equity derivatives:**

- Equity Swaps (TRS, PRS, single name/index/basket, VarSwap, VolSwap, Dispersion, Correlation, Dividend Swap)
- Options & Forwards

- **Foreign Exchange derivatives:**

- FX Swap, Forward, NDF, Options

- **Commodity derivatives:**

- Swaps, options, swaptions

- **Exchange Traded derivatives**

The use of common elements allow for representation of multiple types of products and events in the trade workflow with minimal incremental work. Thus, this coverage list does not represent an exhaustive list of all possible combinations of elements or events

Product & Event Coverage

The scope of contractual products and events in the current model are summarized below:

- **Securities Lending:**
 - Single underlier, cash collateralised, open/term security loan
- **Repurchase Agreements:**
 - Open Term, Fixed Term, Fixed Rate, Floating Rate
- **Events:**
 - Allocation, Re-allocation
 - Cash, Security transfers, DVP settlement
 - Clearing events
 - Compression
 - Increase and decreases/returns
 - Novations- full, partial
 - Terminations- full, partial
 - Renegotiation
 - Reset
 - Execution
 - Stock Split
 - Index Transition
 - Determination of corporate action and credit events

The use of common elements allow for representation of multiple types of products and events in the trade workflow with minimal incremental work. Thus, this coverage list does not represent an exhaustive list of all possible combinations of elements or events

How is CDM different to other data standards such as FpML?

ANALOGY: FINANCIAL DATA MODELS & HUMAN BIOLOGY

THE BIOLOGICAL ANALOGY



DNA / GENES: THE BLUEPRINT

A generic model. Defines structure & function (e.g., 2 arms, 2 lungs, organ processes).



HUMAN BODY: THE FUNCTIONAL SYSTEM

An application running on "the model". Lives, grows, acts.



X-RAY / MEDICAL IMAGING: A SNAPSHOT

Captures the state of the system at a point in time. Used for messaging and storage.

THE FINANCIAL APPLICATION



COMMON DOMAIN MODEL (CDM): THE DATA MODEL

A foundational standard data model. Defines digital life cycle of derivatives & financial products.



FUNCTIONAL SYSTEMS: APPLICATIONS

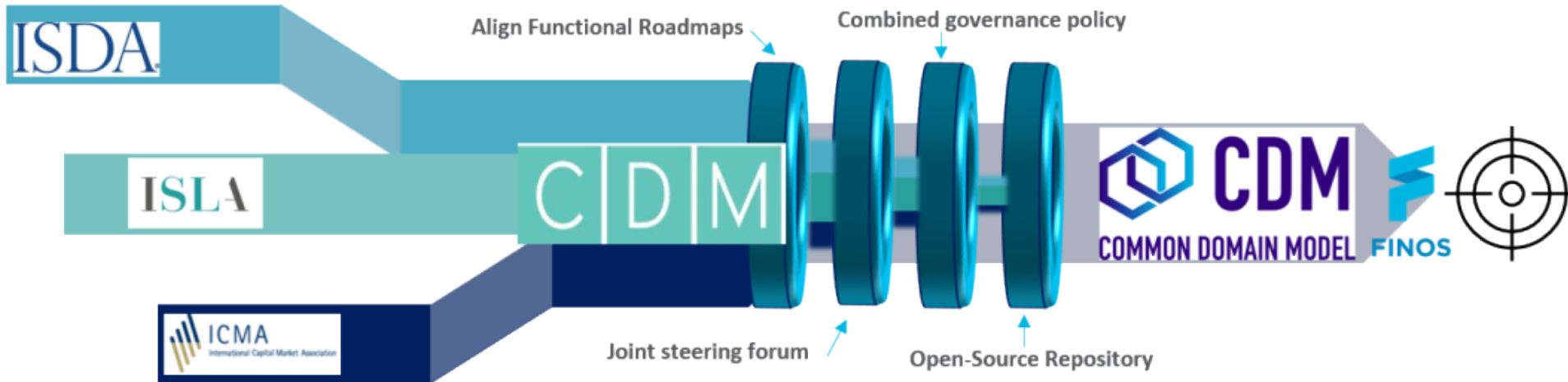
Systems running based on the CDM. Process trades, manage life cycles, calculate risk.



FpML/JSON: SNAPSHOT MESSAGING

Messaging standard used to communicate point-in-time data. For messaging and storage.

Trade Association Collaboration



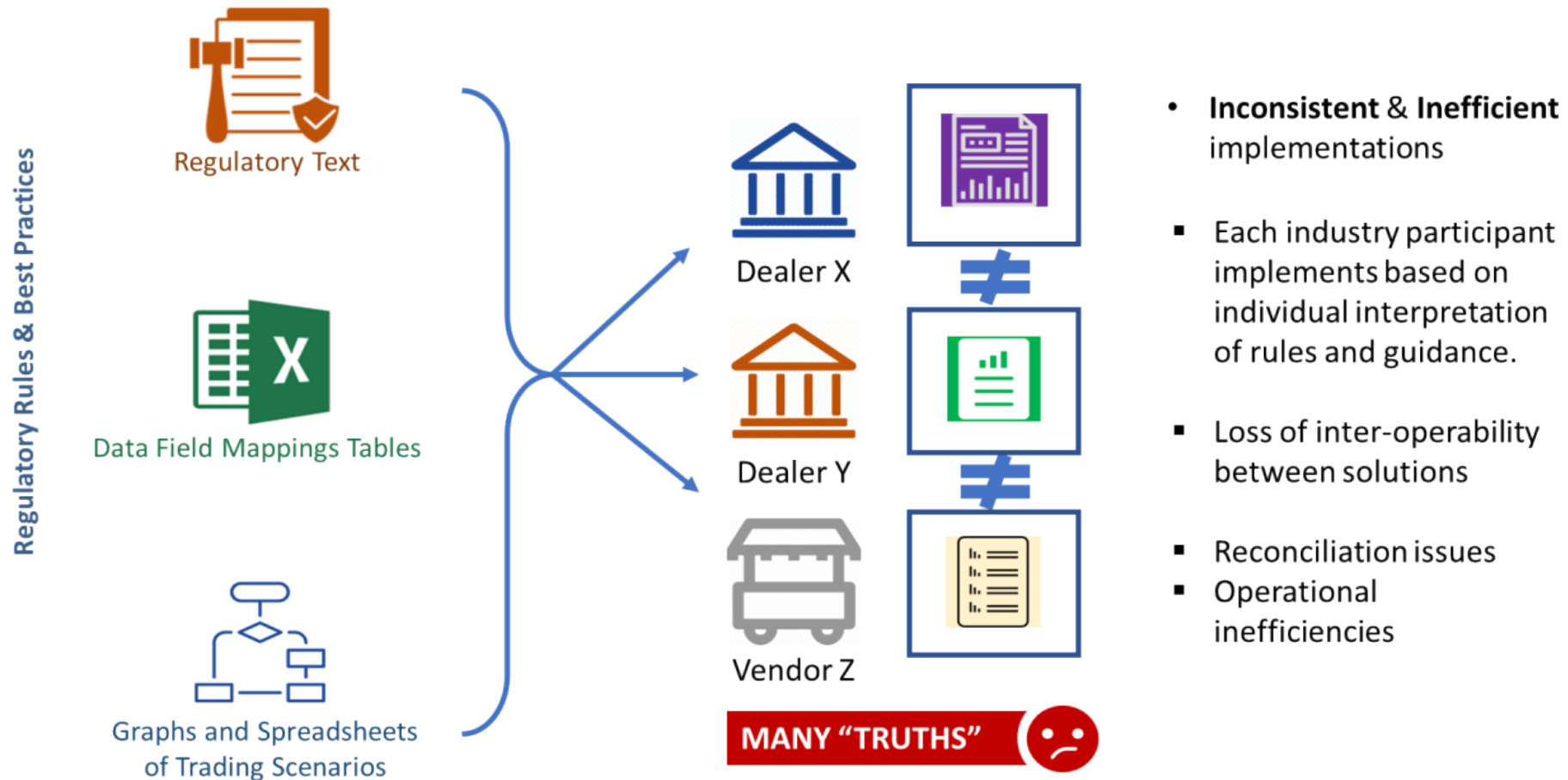
- Best-practice led Governance
- Open-Source Community of Members

Associations are collaborating towards the same future goal, to benefit the whole industry

- An open-source model mutualises cost of development between TAs and contributing firms while retaining best practice governance
- MoU in 2021 enshrined collaboration publicly
- Working groups were opened to each others' members
- TAs appointed FINOS to provide a repository with a view to fostering the growth of an open-source community for the CDM, with migration completed early 2023

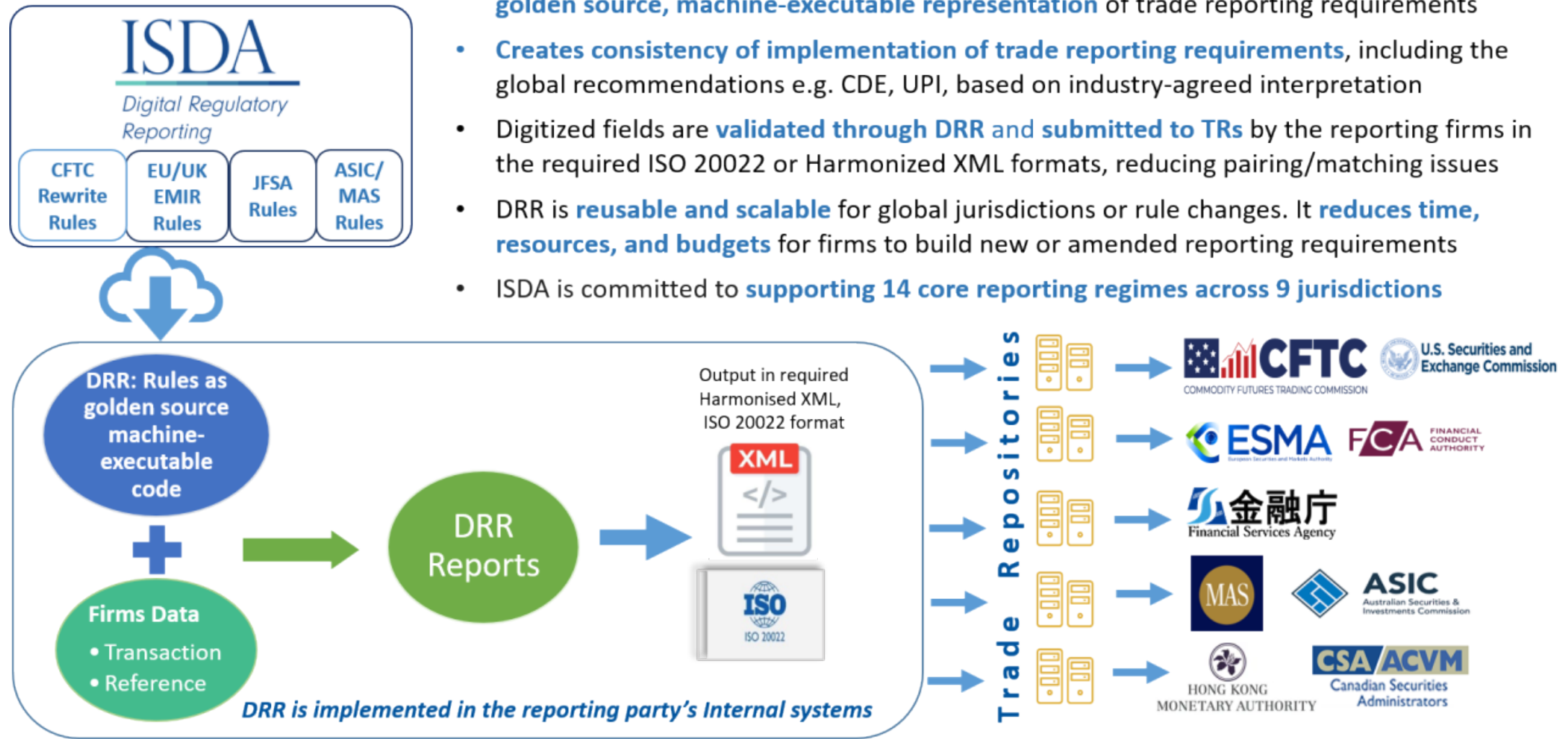
Regulatory Reporting Implementation: The Problem Statement

Industry firms currently build based on individual rule interpretations

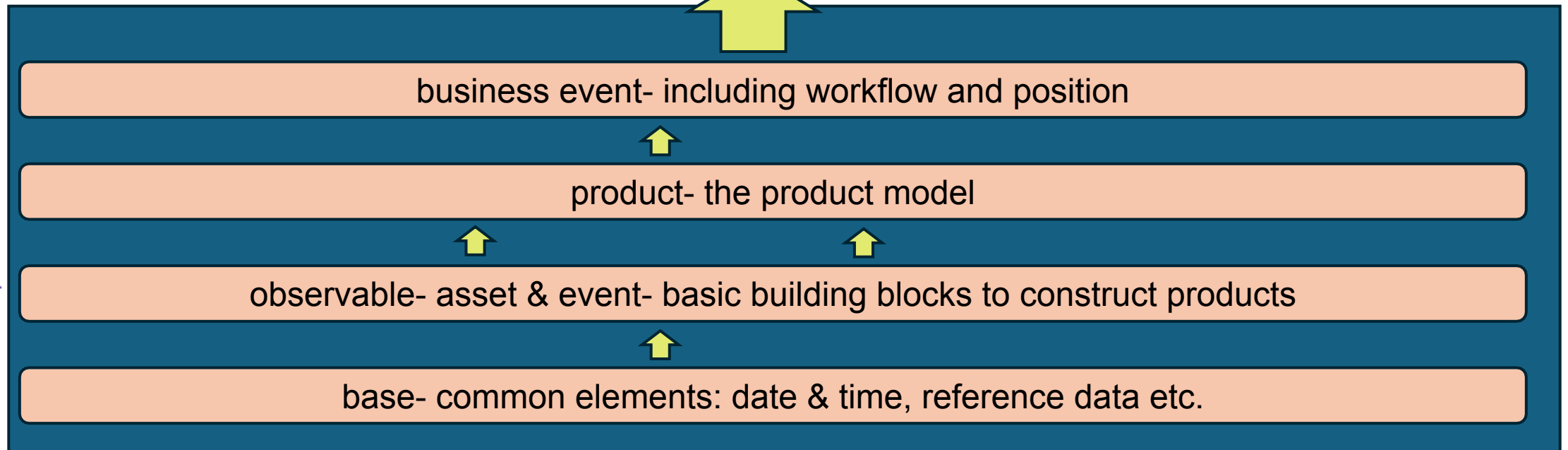
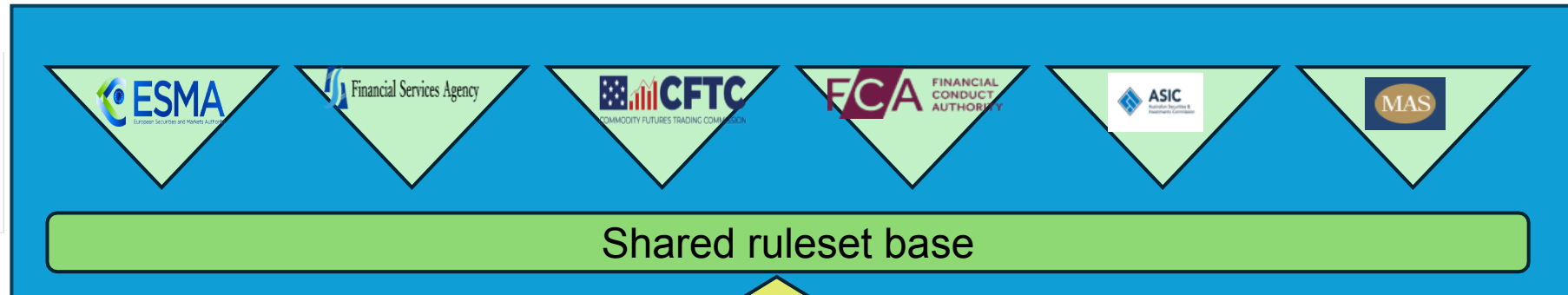


There has to be a better way!

Regulatory Reporting Implementation: Using the ISDA DRR Solution



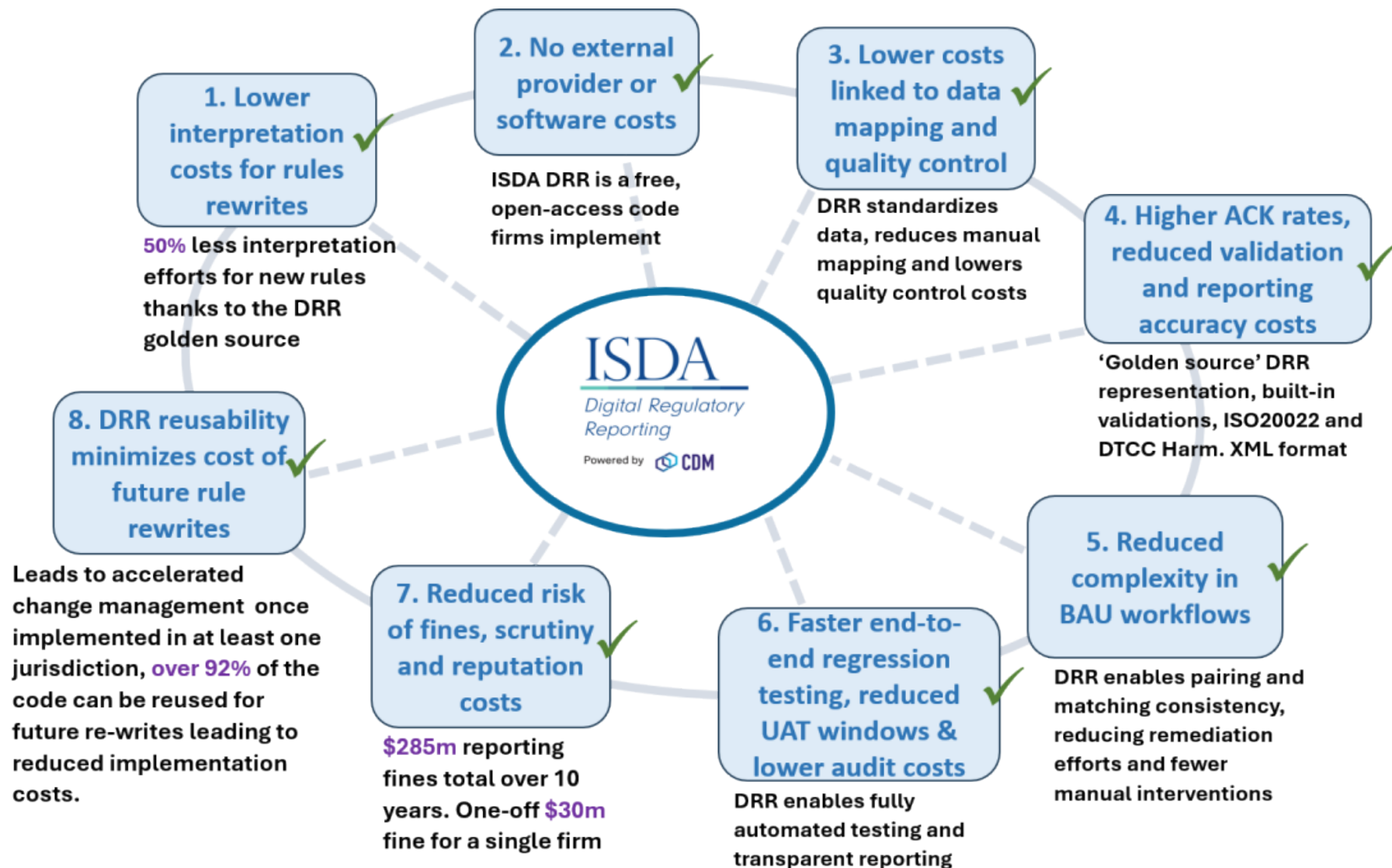
How CDM and DRR work together to produce one solution



Benefits of DRR

ISDA DRR streamlines firms' regulatory reporting, reduces manual effort and errors, and improves compliance in eight key areas.

DRR cuts operational costs and reduces the risk of penalties by delivering consensus interpretation, efficiency, control, and transparency.



DRR: The Value Proposition

Reduces regulatory reporting implementation and maintenance costs and mitigates non-compliance risk

Situation

Global Regulators are amending their reporting requirements to incorporate globally agreed data standards to improve the cross-border consistency of what is reported. However, variations in the requirements, inconsistent industry interpretation and implementation means errors and inconsistencies will continue, resulting in possible regulatory penalties for those firms that get it wrong.

Complication

Multiple regulators plan to implement new or revised rules, but they won't be identical. This means firms can't take the work completed for one jurisdiction and apply it to another. Instead, they need to individually interpret, apply and maintain each rule set, adding to costs, complexity and the burden on resources. Firms may also not interpret the requirements in the same way, leading to errors and the potential for regulatory censure.

Solution

ISDA's Digital Regulatory Reporting (DRR):

- Creates consistency, as it is based on an industry-agreed interpretation of the rules and best practices and validates reporting by firms.
- Uses the Common Domain Model (CDM) to transform that consensus interpretation into unambiguous, machine-executable, open-access code.
- Can be extended to cover new rules and adapted to reflect future rule changes.
- ISDA is committed to supporting 11 core regulatory reporting regimes, across 9 jurisdictions.
- Can be used to generate submission to trade repositories in multiple formats, including ISO20022.

Benefits

Avoids regulatory non-compliance: Regulators globally have issued fines of more than **\$285 million** for incomplete or inaccurate reporting. The **Luxembourg regulator** – the **CSSF** – identified the most common reasons for rejections under EMIR Refit being due to non-compliance with validation rules. **ASIC** also found widespread reporting weaknesses in identifiers, transaction terms, valuations, collateral. The ISDA DRR mutualized industry interpretation reduces this risk by ensuring all firms report the same data requirements in the same way.

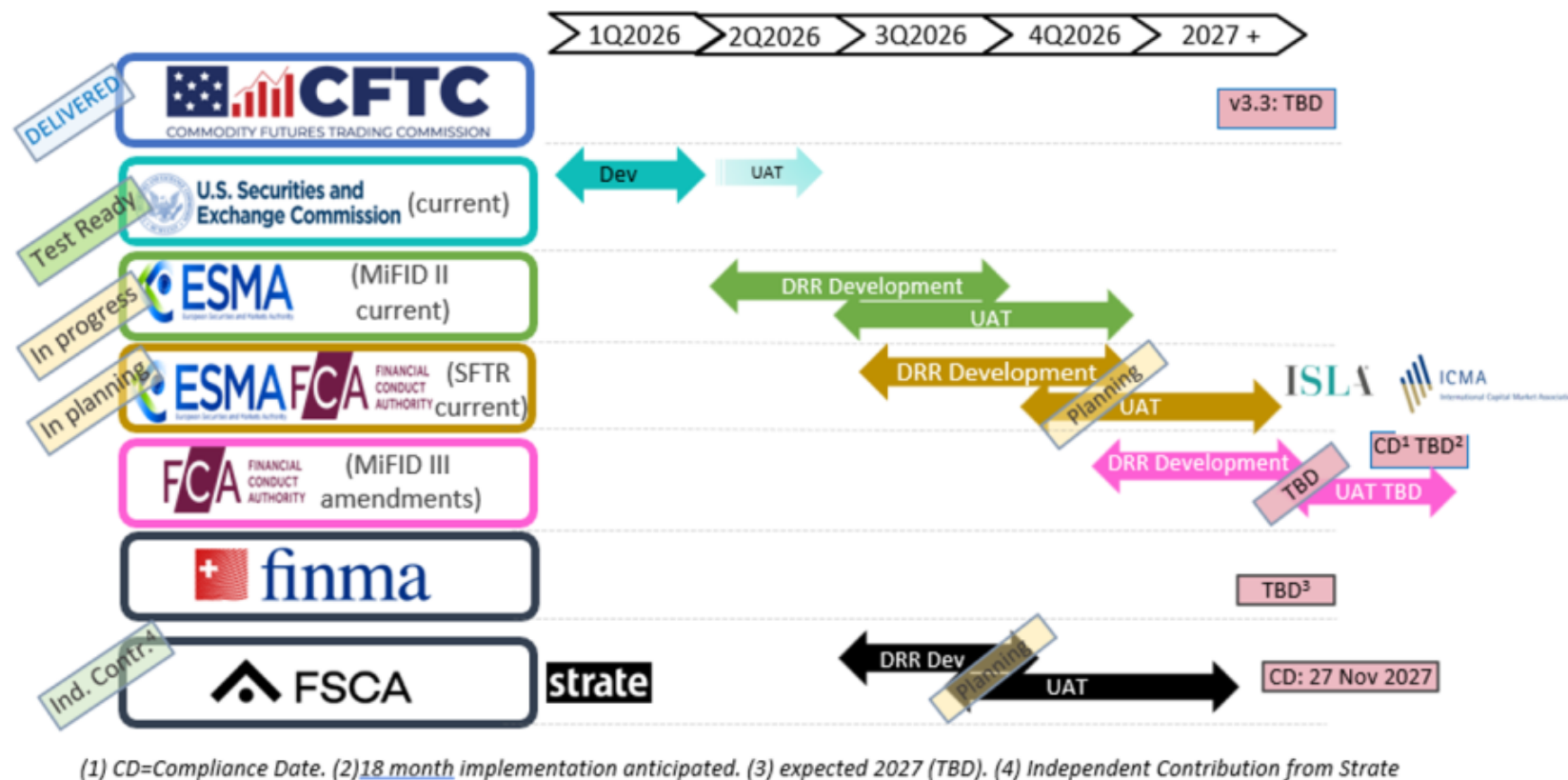
Reduces costs: Using free machine-executable code produced from an industry mutualized approach and updated over time avoids duplication of effort across multiple industry participants. This enables fewer resource requirements per firm and allows resources to be reduced or reallocated to more value-add tasks.

DRR Deliveries So Far

Regulation	Implementation Date	ISDA DRR Status
CFTC	December 5, 2022	✓ Delivered
	CFTC 3.2: March 2023	
JFSA	Phase 1: April 1, 2024	✓ Delivered
	Phase 2: April 7, 2025	
ESMA EMIR	April 29, 2024	✓ Delivered
FCA EMIR	September 30, 2024	✓ Delivered
ASIC	October 21, 2024	✓ Delivered
MAS	October 21, 2024	✓ Delivered
CSA	July 25, 2025	✓ Delivered
HKMA	September 29, 2025	✓ Delivered

Digital Regulatory Reporting Delivery Roadmap 2026+

- **Deliver key 2026 regulatory priorities**, including SEC and advancing MiFIR (EU/UK) towards 2027 readiness.
- **Progress modelling of additional jurisdictions**, including South Africa, and prepare for future expansions in the DRR roadmap.
- **Advance SFTR modelling** in collaboration with industry partners to support broader cross-jurisdiction alignment.
- **Strengthen traceability and transparency** across DRR workflows by maturing end-to-end traceability capabilities.
- **Maintain BAU support** for existing DRR jurisdictions while scaling capacity for new ones.
- **Promote industry adoption** by supporting members in onboarding, integration and readiness activities.

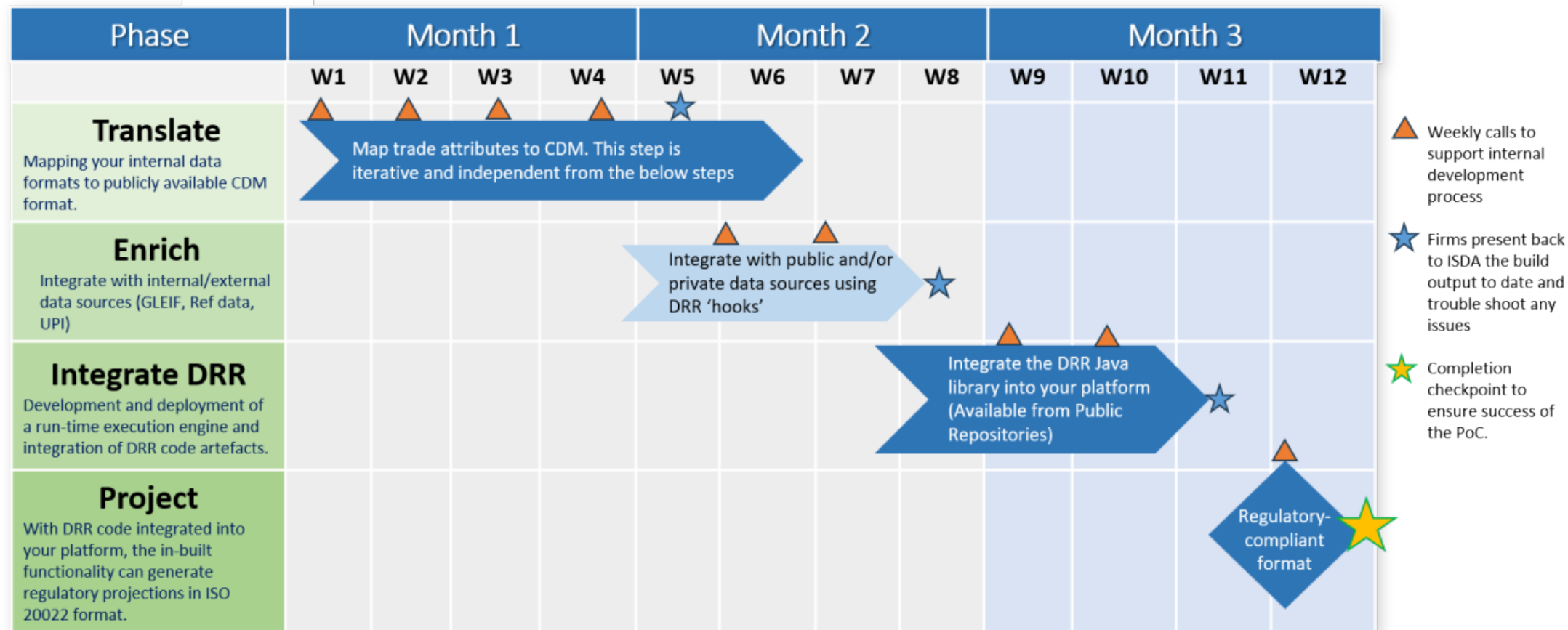


DRR PoC Timeline- High Level

GETTING STARTED:



- ✓ Identify 1 Derivative Product. Other firms have used FX Forwards, Equity Derivatives, CDS/IRS for their PoC
- ✓ Identify 1 jurisdiction for the regulatory transaction reporting output. You can compare the output of the PoC to your existing reporting output
- ✓ Start with a limited scope of 'New' Transactions initially, subsequently include 'Amendments' 'Events' etc.
- ✓ To scale to other products/asset classes or event types, expand the CDM mapping in step 1 & 2
- ✓ To support additional regimes, update your DRR configuration by simply calling the additional DRR jurisdiction code



Strate Contribution

- Overview of the South African regulatory landscape
 - FMA Conduct Standard 3 (CS3) published October 2018
 - Scope:
 - Entities: OTC Derivative Providers (ODP)
 - Products: OTC derivatives – all 5 asset classes
 - Events: Lifecycle event reporting (execution, termination, modifications)
 - Reportable data fields will be based on EMIR reporting (consultation paper pending)
- Strate's regulatory reporting responsibilities and priorities
 - The *only* Trade Repository for CS3 reporting
 - Perform validations on submitted messages, provide reports to ODP's, and supply reported data to regulators.
 - Strate also engage with the market, providing a pivotal role in the development of the reporting requirements
- Current thinking and plans around alignment with DRR and future integration
 - Publish and maintain CS3 reporting requirements as machine executable code
 - Available as open access model for entities to adopt – consistent and complete integration of CS3 rules

Adoption Models and Use Cases



DRR Governance Landscape

DRR Steering Group Chair: David Shone, ISDA

- Defines and steers the DRR roadmap, including priorities, scope and delivery timelines
- Provides strategic direction, industry level oversight, and advisory input to shape the DRR initiative across the global derivatives market
- Monitors overall roadmap, setting annual priorities
- Participation is determined by the eligibility criteria as set out in the DRR Steering Group Terms of Reference
- Outcomes and recommendations are reported through ISDA's internal program governance framework
- Frequency: Monthly meetings

DRR Steering Group (DRR SG)*

Initiative Proposals,
Updates, Escalation of
Critical Issues

Direction, Guidance,
Validation, Approvals

DRR Working Group (DRR WG)

Changes to CDM
impacting DRR

Regulatory/Policy/Rule
Interpretations

Changes
required to
trades/lifecycle

DRR Working Group Chair: Tabish Ahmed, ISDA

- Modelling, design and execution of work items
- Ensures contributions are compliant and consistent across DRR in line with DRR WG Terms of Reference
- Feeds domain-based work items and plans into DRR roadmap
- Assembles releases, validates completed code, DRR user queries triage/ discussion
- Interacts with ISDA Data & Reporting Industry WGs and FINOS CDM Domain WGs and sub-groups, as required
- Frequency: Fortnightly meetings

ISDA Data & Reporting Industry Working Group Landscape

ISDA Data & Reporting EMEA Working Group Chair: Andrew Bayley, ISDA

ISDA Data & Reporting U.S. Compliance Working Group

Chair: Eleanor Hsu, ISDA

ISDA Data & Reporting Canadian Group Chair: Eleanor Hsu, ISDA

ISDA Japan Data & Reporting Compliance Chair: Keisuke Nakamura, ISDA

AEJ Reporting Working Group Chair: Eleanor Hsu, ISDA

ISDA South Africa Reporting Working Group Chair: Andrew Bayley, ISDA

FINOS CDM & DWG Governance Landscape

CDM Steering Group

CDM SG

Chair: David Shone, ISDA
Meets Monthly

Contribution Review Working Group (CRWG)

CRWG

Chair: rotated between TAs
Meets Fortnightly

Technology Architecture Working Group (TAWG)

TAWG

Chair: Chris Rayner, ISLA
Meets Monthly

CDM Domain Working Groups

CDM DWGs

Chair: N/A
Meets Ad-hoc

DWGs that use ISLA governance & infrastructure

ISLA CDM WG

Chair: Chris Rayner, ISLA
Meets Monthly

DWGs that use ICMA governance & infrastructure

ICMA Implementation WG

Chair: Tom Healey, ICMA
Meets as needed

* Outcomes/ recommendations from the DRR SG are reported up through ISDA's internal program governance framework

DRR Adoption Status Overview

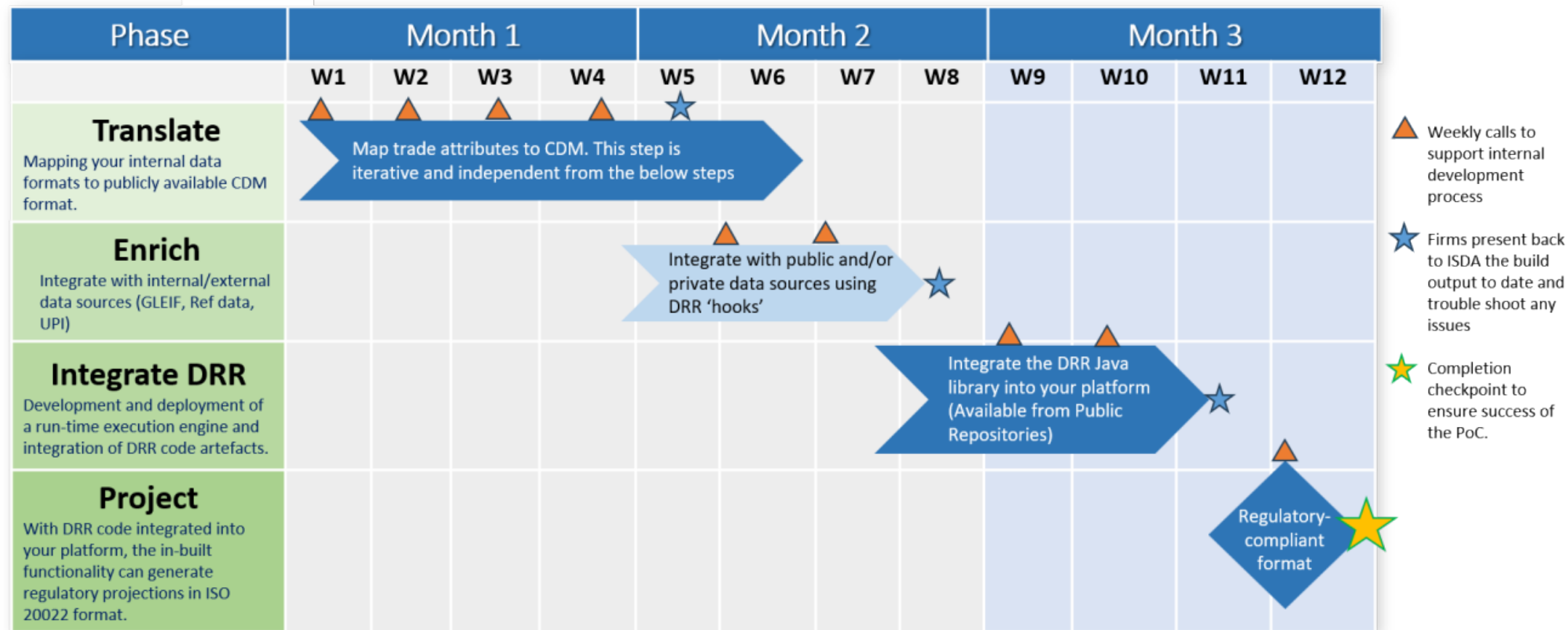
- Significant increase in ISDA DRR adoption across regions in 2025. South Afr. STRATE TR will publish new rules as code in DRR
- Firms currently using DRR for reporting such as JP Morgan, JSCC, LSE, Natixis, Pictet actively promote DRR to peers.
- Positive feedback from EMIR Refit and UK EMIR Refit rules go-live
 - Members who adopted CDM/DRR achieved higher ACK rate than the industry average:
 - DTCC noted that the proportion of successfully submitted reports during the first week of the revised EMIR reporting rules in the EU was roughly 75%, climbing to 96% after two months, whereas DRR users reported **close to complete accuracy** from day one, reducing the risk of regulatory penalties for incorrect reporting. Early DRR adopters shared they were **glad they made the strategic decision to leverage CDM/DRR for reporting** as various jurisdictions roll out the rewrite rules.
 - Firms using DRR for their EMIR, MAS, ASIC, Canada and HKMA reporting were able to start UAT before the majority of the market. In fact, ISDA had completed DRR before Trade Repository UAT. ISDA plans for this to be recurring theme for future implementations, e.g. CFTC & MiFIR.
- ISDA has completed Canada and Hong Kong rewrites in 2025, with both regimes now live in DRR, while SEC functionality has been delivered and is available for testing; this is being reinforced through **active member participation in peer review**, a **public JPMorgan implementation webinar**, and **regular CDM/DRR training by ISDA** across major financial centres
- DRR adoption continues to build momentum, with 7 firms now live: JPMorgan, Banque Pictet, BNP Paribas, JSCC, Natixis, LSEG and 1 TBA
- JSCC announced DRR adoption in July 2025 for JFSA and CFTC, becoming the first CCP and first Japanese entity to adopt DRR in production
- LSEG announced DRR adoption in 2026 by integrating CDM and DRR into its TradeAgent post-trade service
- Pipeline activity is strong, with 15 firms conducting DRR PoCs and 25 other firms currently at DRR technical analysis stage
- Firms currently undertaking PoCs include Bank of America, Santander, MUFG, SMBC, STRATE, DTCC, Novatus Global and Broadridge, **underlining broad interest across banks, market infrastructures and service providers**

DRR PoC Timeline

GETTING STARTED:



- ✓ Identify 1 Derivative Product. Other firms have used FX Forwards, Equity Derivatives, CDS/IRS for their PoC
- ✓ Identify 1 jurisdiction for the regulatory transaction reporting output. You can compare the output of the PoC to your existing reporting output
- ✓ Start with a limited scope of 'New' Transactions initially, subsequently include 'Amendments' 'Events' etc.
- ✓ To scale to other products/asset classes or event types, expand the CDM mapping in step 1 & 2
- ✓ To support additional regimes, update your DRR configuration by simply calling the additional DRR jurisdiction code

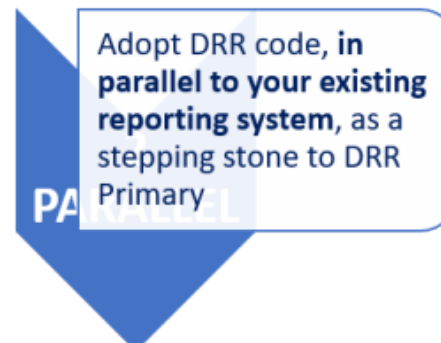


Different Forms of Adoption

DRR AS BENCHMARK



BUILD DRR INTO YOUR SYSTEM



VALUE

- DRR is a **mutualized community solution**. It is the industry's **common interpretation of reporting requirements** and offers a **maintained reporting benchmark** for firms
- Human readable '**golden source**' ensuring **alignment with industry consensus**
- Offers confidence in **reporting accuracy**
- Serves as a **control or corroboration** of firms' reporting output to trade repositories
- Can be used by vendors for **accuracy check** services
- **Traceability** of reported data back to the 'golden source'
- **Can be used by vendors as a 'utility'** supporting their services particularly **reporting software** and **accuracy checks**
- **Delegated reporting services** aligned with industry standards

IMPLEMENTATION

- Light, may require development of translation from firms' data formats to those supported by publicly available CDM & DRR
- Free access to Rosetta community edition needed for DRR and CDM and/or connection via API
- Implemented either directly through internal integration effort or through use of a third-party vendor who has adopted the DRR code. Requires adoption of the Common Domain Model (CDM) for data staging
- Development and deployment of a run-time execution engine, integration of DRR code artefacts, translation from internal data formats to CDM, own code generator based on publicly available formats (distributed with CDM/DRR)
- Partial adoption of CDM as Proof-of-Concept
- Adoption of the CDM for data staging (natively or via vendor)

Realised Adoption Benefits

- DRR is designed to deliver benefits:
 - Cost efficiency
 - Standardisation and control
 - Regulatory consistency and transparency
- ISDA and Capgemini published a joint report in November 2025 titled *Industry Perspectives on the ISDA DRR: Unlocking Efficiency, Accuracy, and Strategic Value*, examining market adoption of the Digital Regulatory Reporting (DRR) initiative
- The next few slides outline the key findings from that report

Improved Accuracy & Reduced Complexity

- Digitized regulatory rules and built-in validations drive higher first-time accuracy
- 100% ACK rate under MAS rules during the go-live period
- 98.2% ACK rate following EMIR Refit implementation — both above established BAU benchmarks
- Fewer manual interventions mean faster root-cause analysis when issues do arise

100%

ACK Rate — MAS Rules (go-live period)

98.2%

ACK Rate — EU EMIR Refit

Streamlined Change Management & Cost Efficiency

- Reusable code and logic accelerate the change-management process across jurisdictions
- One firm reused 85% of its code across jurisdictions during rewrites
- Another reused 92% of existing DRR code during its CSA (Canada) implementation
- Reported cost reductions of up to 50% since implementation
- Most firms delivered within existing regulatory change budgets — no separate DRR investment line required

CODE REUSE ACHIEVED



50%

Reduction in ongoing regulatory maintenance costs reported since implementation

Operational & Strategic Benefits

1

Standardization across jurisdictions

Reduces fragmentation and regulatory interpretation risk across EMIR, MAS, CSA, HKMA, ASIC and JFSA

2

Externalized issue management

ISDA and FINOS working groups increasingly absorb BAU troubleshooting, freeing internal operations capacity

3

Faster remediation cycles

Developers can run comprehensive end-to-end validation independently, shortening UAT windows

JURISDICTIONS COVERED BY ISDA DRR

EU (EMIR)

UK (EMIR)

US (CFTC)

Singapore (MAS)

Hong Kong (HKMA)

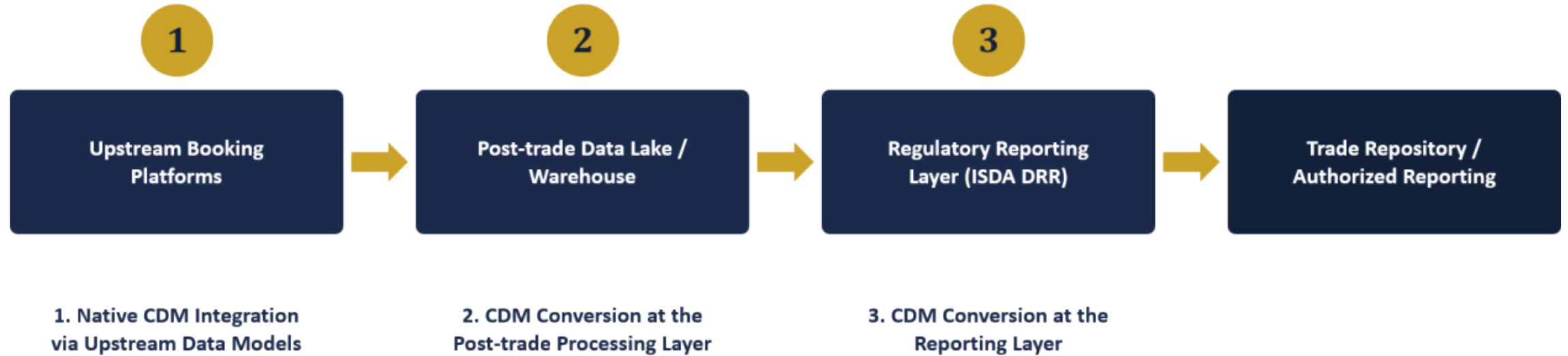
Canada (CSA)

Australia (ASIC)

Japan (JFSA)

Three Paths to CDM Integration

Firms choose where in the data stack to integrate the CDM, based on cost, complexity and operational priorities



Each option reflects a different balance of implementation cost, speed to deploy and long-term operational value — detailed on the following pages.

1) Native CDM Integration via Upstream Data Models

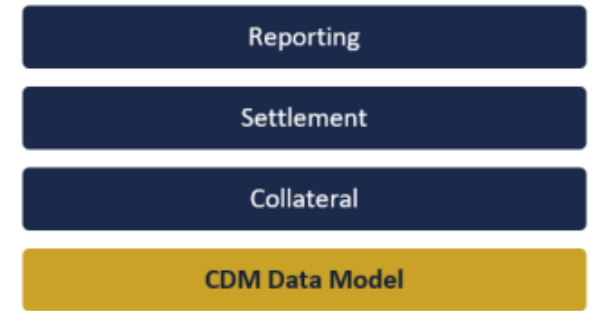
CDM adopted as the native data model across the full trade lifecycle — from trade capture through post-trade and reporting

ADVANTAGES

- Full lifecycle standardization and straight-through processing (STP)
- Enhanced data quality from upstream, via integrated validation
- Enables machine-executable regulatory reporting via ISDA DRR
- Unlocks broader CDM use cases: settlement, collateral, legal docs

TRADE-OFFS

- Higher upfront investment in system redesign and training
- Requires robust governance and architectural decision frameworks
- Longer implementation timelines — typically 6–8 months after POC



Best suited for: Firms seeking long-term scalability, cross-jurisdictional compliance, or planning to replace legacy booking systems.

2) CDM Conversion at the Post-trade Processing Layer

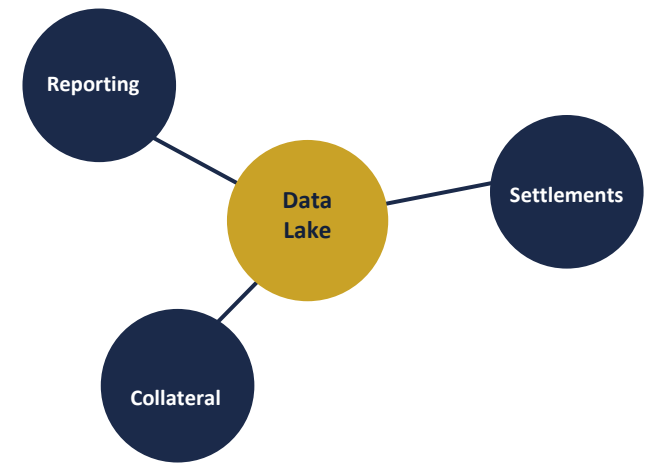
CDM implemented within the data lake or warehouse, creating a centralized repository that feeds multiple post-trade processes

ADVANTAGES

- Integrated data validation in the data lake enhances data quality
- Facilitates automation of downstream post-trade processes
- Minimal changes required to existing trade capture systems

TRADE-OFFS

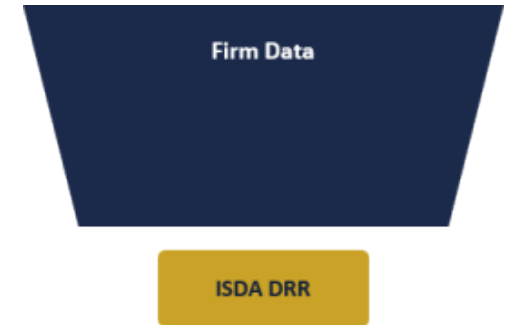
- Higher upfront investment in system architecture design vs. reporting-layer conversion
- Still requires robust governance and architectural decision frameworks



Best suited for: Firms wanting to utilize the CDM across post-trade processes with minimal upstream changes and efficient implementation.

3) CDM Conversion at the Reporting Layer

Existing internal or FpML data formats are converted into CDM-compliant structures only at the point of regulatory reporting



ADVANTAGES

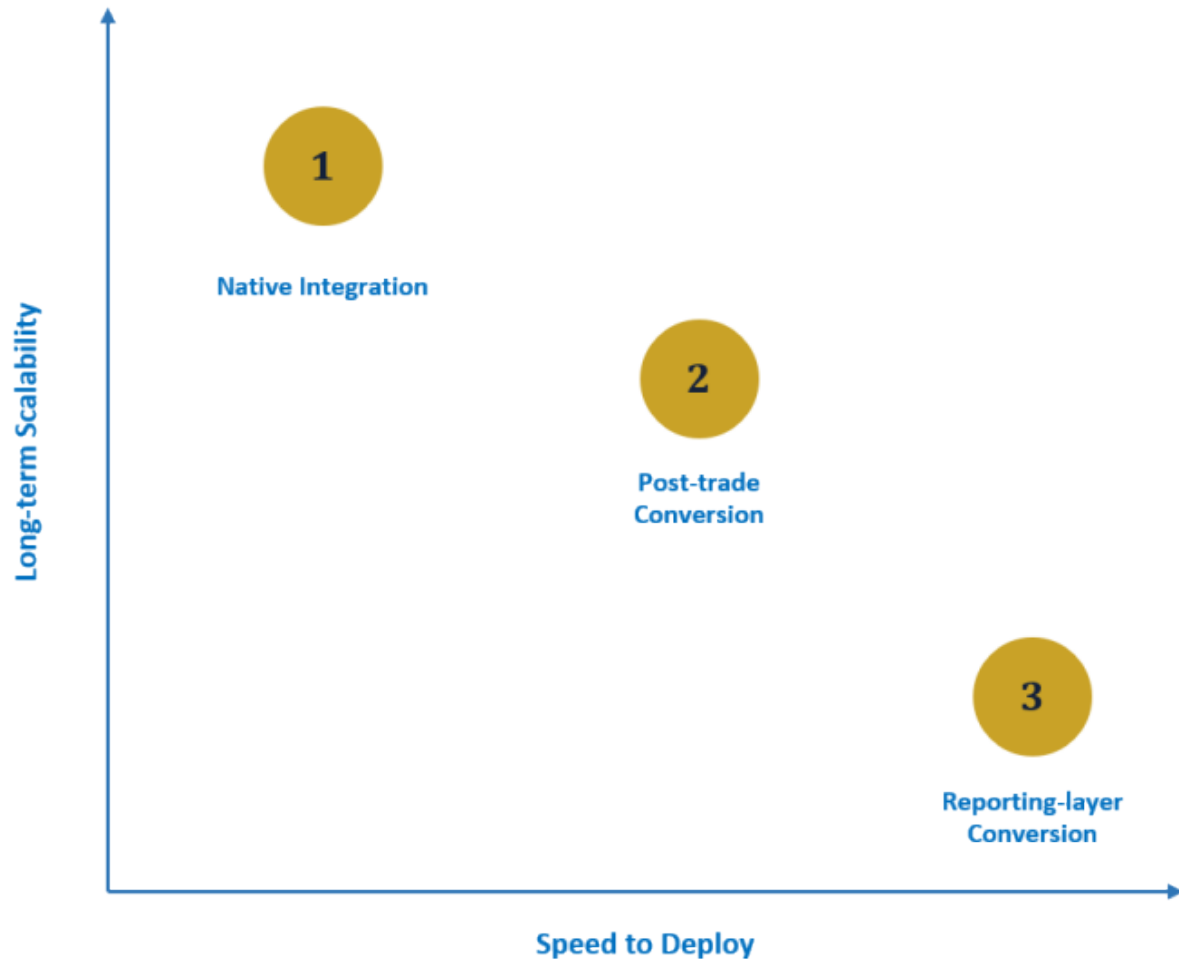
- Minimal disruption to upstream and data lake systems
- Fastest time-to-market for ISDA DRR compliance
- Easier to pilot via POCs with limited scope (eg, interest rate products)

TRADE-OFFS

- Limited scalability across asset classes and jurisdictions
- Potential duplication of logic and mappings across systems
- Less opportunity to benefit from full lifecycle standardization

Best suited for: Firms with a mature reporting infrastructure but fragmented upstream data models that are seeking faster implementation.

Choosing the Right Option



All three approaches proved viable in practice. The right choice depends on a firm's priorities — speed vs. scalability, cost tolerance and existing system architecture.

Most firms began with a 3-month POC on a single product and jurisdiction — commonly FX forwards, equity derivatives, credit default swaps or interest rate swaps — regardless of the ultimate integration strategy chosen.

12–21 months for full implementation; adding a new jurisdiction after go-live typically took **under 3 months**.

South African Considerations

- Considerations for South African firms:
 - Scope: identify the entities, products, and events that are in-scope for CS3 reporting.
 - Data: what data must be submitted? Is all the required data currently captured and available? Does additional data need to be collected and/or derived?
 - Message schema and submission: the required format for transaction messages, and submission of the message to the TR
 - Controls and reconciliation: sufficiently robust and reliable internal controls are required to ensure reporting accuracy and completeness. Processes in place to resolve errors as and when identified.
- Cross-border reporting
 - Transactions may be in-scope for multiple jurisdictions
 - With no cross-boarder sharing of data, each regime must be considered, and processed, as separate
 - *NB. ISDA are working on how regulators can effectively share data, based on the CDM*
- Scalability as regimes evolve
 - Every global regime currently in place has experience changes from minor updates and full-scale rewrites
 - Every update requires regulatory interpretation, technical builds, testing and implementation
 - *Implement once* using a centrally published model (DRR). Updates to rules implement once via DRR – reduce time and costs to the industry.

Networking Break



Business Q&A and Discussion



Lunch



Building Transactions with the CDM



CDM Core Concepts from a Technical Perspective

CDM is distributed as open-source code, managed under FINOS and is implemented using the Rune Domain Specific Language (DSL) also managed under FINOS.

CDM defines both:

- Data structures (what a trade looks like)
- Behaviour and functions (how trades change over time)

So, CDM is effectively an event-driven state machine for financial transactions.

A transaction exists in a defined TradeState. Business events such as execution, allocation, amendment, novation, compression, or settlement trigger state transitions, producing new TradeStates while preserving full lineage and traceability.

This allows lifecycle events to be executed consistently across platforms.

CDM Core Concepts- Rune DSL

The CDM data model is defined through the following fundamental entities:

1. **Types:** A data type describes a logical concept of the business domain being modelled. There are two kinds of types:
 - a. **Basic types:** deemed fundamental and applicable to any model. It includes the following:
 - int - integer numbers
 - number - decimal numbers
 - boolean - logical true or false
 - string - text
 - time - simple time values, e.g. "05:00:00"
 - a. **Complex types:** they are specified through a set of attributes defining the granular elements composing them. They are defined by the “type” keyword followed by the **type name**, **name of the attributes**, **type of the attributes**, **cardinality of the attributes**, and may contain **plain-text descriptions**.

attributes {
type Party: <"A class to specify a party.">
partyId PartyIdentifier (1..*) <"The identifier associated with a party.">
name string (0..1) <"The party name.">
businessUnit BusinessUnit (0..*) <"Optional organization unit information used to describe the organization units (e.g. trading desks) involved in a transaction or business process, incl. the contact information (when relevant).">
person NaturalPerson (0..*) <"The person(s) who might be associated with the party as part of the execution, contract or legal document.">
personRole NaturalPersonRole (0..*) <"The role of the person(s) ">
account Account (0..1) <"The account that might be associated with the party.">

CDM Core Concepts- Rune DSL

2. Enumerations: Enumeration is the mechanism through which a type may only take some specific controlled values. An enumeration is the container for the corresponding set of controlled values. They are defined by the “enum” keyword followed by the **enumeration name**, **values**, and may contain **plain-text descriptions**.

values

```
enum NaturalPersonRoleEnum: <"The enumerated values for the natural person's role.">
  Broker <"The person who arranged with a client to execute the trade.">
  Buyer <"Acquirer of the legal title to the financial instrument.">
  DecisionMaker <"The party or person with legal responsibility for authorization of the execution of the transaction.">
  ExecutionWithinFirm <"Person within the firm who is responsible for execution of the transaction.">
  InvestmentDecisionMaker <"Person who is responsible for making the investment decision.">
  Seller <"Seller of the legal title to the financial instrument.">
  Trader <"The person who executed the trade.">
```

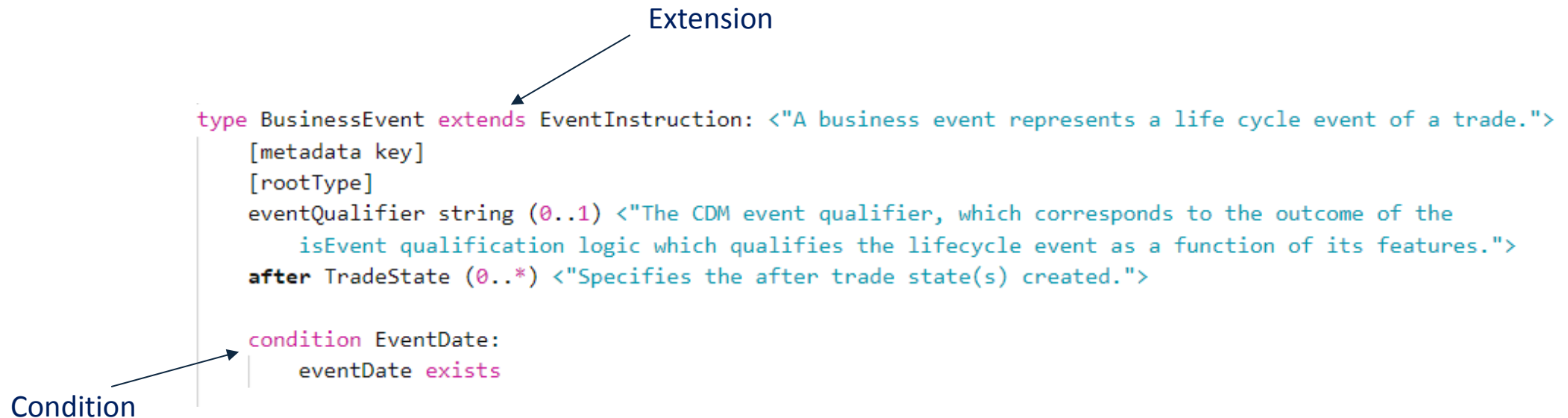
The NaturalPersonRole type contains a complex-type attribute and an enumeration.

```
type NaturalPersonRole: <"A class to specify the role(s) that natural person(s) may have in relation to the contract.">
  personReference NaturalPerson (1..1) <"A reference to the natural person to whom the role refers to.">
    [metadata reference]
  role NaturalPersonRoleEnum (0..*) <"FpML specifies a person role that is distinct from the party role.">
    [metadata scheme]
```

CDM Core Concepts- Rune DSL

Type extensions: Types can also be defined as an extension of another already-defined type. In that case, the type will inherit the attributes of the one that it is extending.

Type conditions: Conditions are logic expressions associated to a data type. They are predicates on attributes of objects of that type that evaluate to True or False. As part of the object validation process, all the conditions are evaluated and if any evaluates to false then the validation fail.



```
type BusinessEvent extends EventInstruction: <"A business event represents a life cycle event of a trade.">
  [metadata key]
  [rootType]
  eventQualifier string (0..1) <"The CDM event qualifier, which corresponds to the outcome of the
    isEvent qualification logic which qualifies the lifecycle event as a function of its features.">
  after TradeState (0..*) <"Specifies the after trade state(s) created.">

  condition EventDate:
    eventDate exists
```

CDM Core Concepts- Rune DSL

Functions: functions are a fixed set of logical instructions returning an output which can be parameterised by a set of inputs (also known as arguments). They transform the input data letting us to reply and automate complex business processes.

They are defined by the “func” keyword followed by the **function name**, **input values** (name, type, and cardinality), **output** (name, type, and cardinality), **output construction statements**, and may contain **plain-text descriptions**.

```
func FilterCashTransfers: <"This function returns the cash transfers containing a currency.">
  inputs:
    | transfers Transfer (0..*) <"Cash transfers to filter">
  output:
    | cashTransfers Transfer (0..*) <"Cash transfers that contain a currency.">
  add cashTransfers:
    | transfers filter quantity -> unit -> currency exists
```

The **`add`** keyword allows to append to the output list (cashTransfers) the values returned by the expression.

In the other hand, the **`set`** keyword can be used to override the entire list with the value returned by the expression, which also needs to be a list in this case.

Functional expressions are composable, so a function can call another function.

CDM Core Concepts- Rune DSL

Namespace definition: Namespaces are defined at the beginning of a file by the “namespace” keyword followed by the location of the namespace in the directory structure.

Namespace import: The names of all the components (types, enumerations and functions) must be unique across a namespace. Components can refer to components outside their namespace by importing their namespace into the current file, this is done through the “import” keyword followed by the location of the namespace in the directory structure. The asterisk at the end indicates that all the objects of the namespace are imported.

```
namespace drr.regulation.fsca.zaemir.refit.trade
version "${project.version}"

import cdm.base.*
import cdm.base.staticdata.asset.common.*

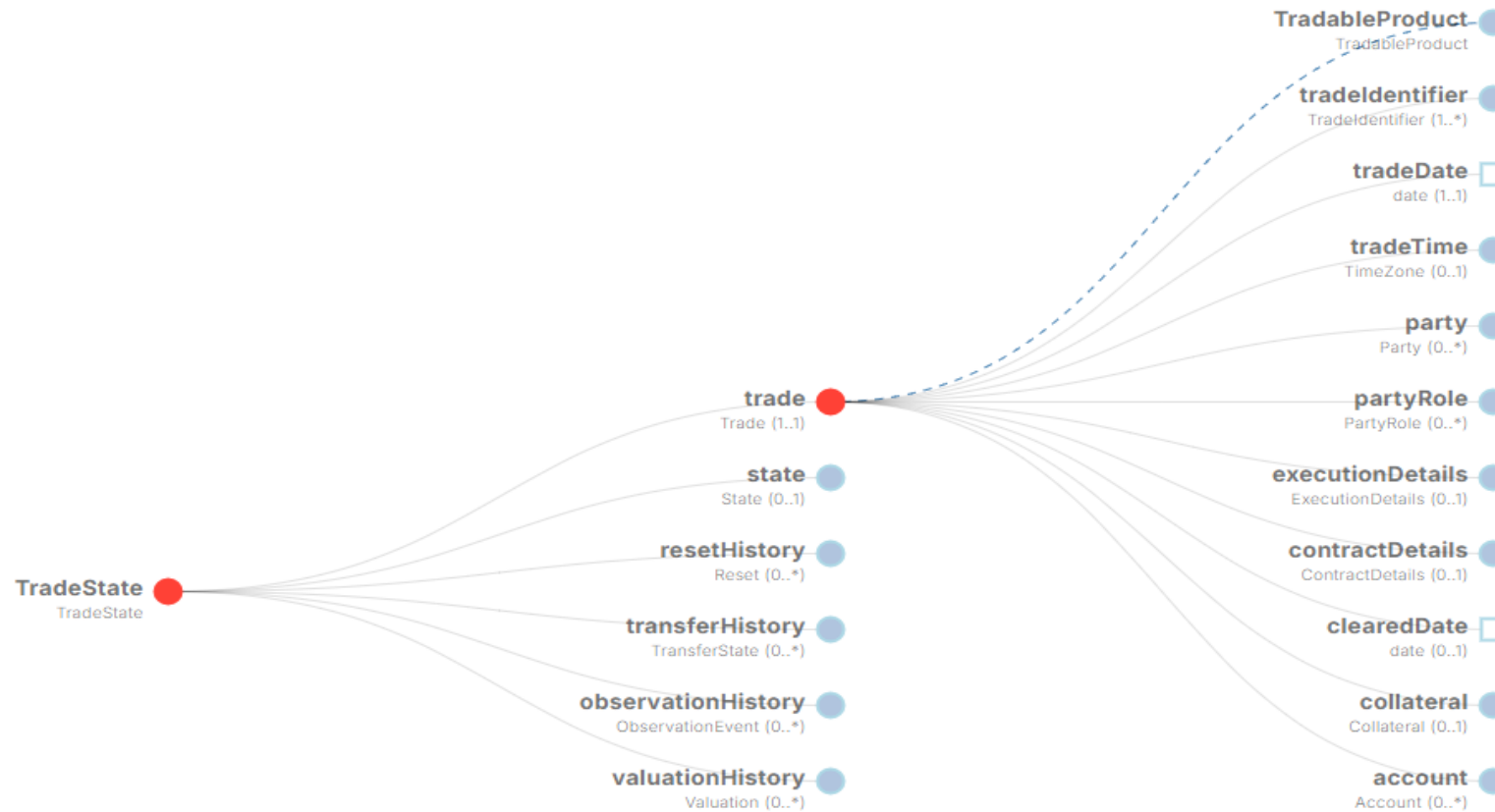
import drr.base.*
import drr.base.qualification.event.*
import drr.base.trade.*
import drr.base.util.datetime.*
import drr.base.util.party.*
import drr.base.util.string.*
import drr.enrichment.eic.*
import drr.enrichment.lei.*
import drr.regulation.common.*
import drr.regulation.common.emir.*
import drr.regulation.common.trade.* as common
import drr.regulation.esma.*
import drr.regulation.esma.emir.*
import drr.regulation.fsca.zaemir.refit.*
import drr.regulation.fsca.zaemir.refit.trade.* as esma
import drr.standards.iso.*
```

```
type FSCAZAEMIRTransactionReport extends EMIRTransactionReport:
```

Construction of CDM Transactions and Lifecycle Events

TradeState:

- Defines the fundamental financial information that can be changed by any life-cycle event.
- It is the input and output of any state transition - with each event, a new TradeState instance is created.



Construction of CDM Transactions and Lifecycle Events

Instruction:

Contains the information that it is used by the *Primitive Functions* to perform the next TradeState(s) after an event has occurred.



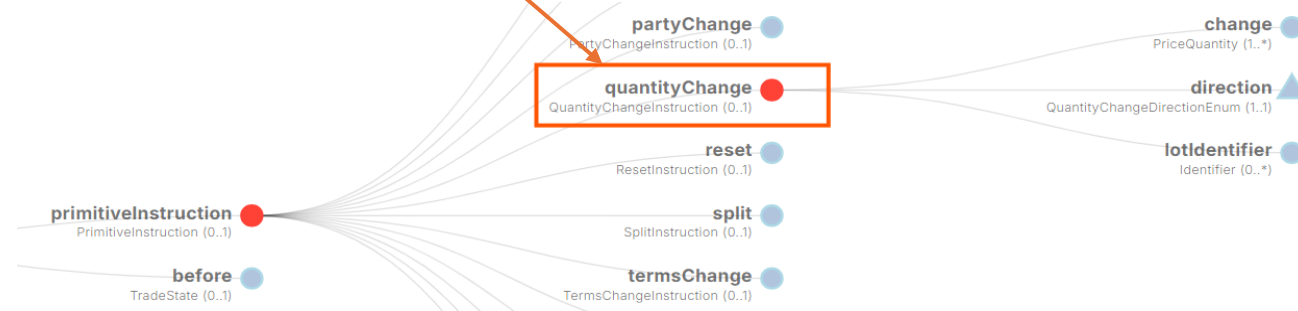
Each primitive function has an associated primitive instruction attribute representing the function's inputs.

By populating more than one attribute, the PrimitiveInstruction data type allows to build composite primitive instructions.

Functions are “chained” by using the after state output of one function as the before state input into the next function. The *Create_TradeState* function performs such composition of primitive functions.

```

func Create_QuantityChange: <"A specification of the input
Primitive Event">
inputs:
  instruction QuantityChangeInstruction (1..1)
  tradeState TradeState (1..1)
output:
  quantityChange TradeState (1..1)
  
```



Construction of CDM Transactions and Lifecycle Events

BusinessEvent:

The Business Event represents the **current life cycle event** of a trade. It contains the resulting **TradeState** after an event has occurred, as well as the instructions that generate it.

The BusinessEvent is created by the *Create_BusinessEvent* function provided in the model.

A business event is atomic, i.e. its primitive components cannot happen independently. They either all happen together or not at all.



Construction of CDM Transactions and Lifecycle Events

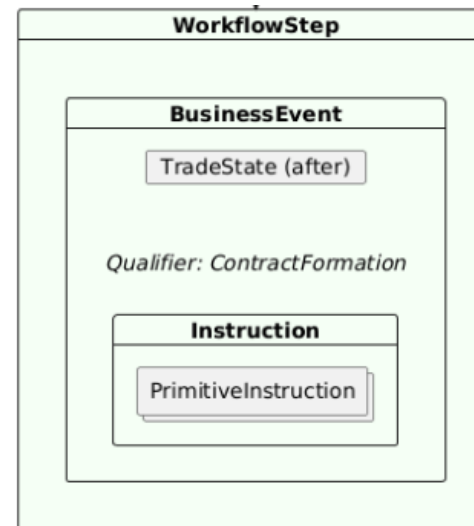
The CDM event model, used to represent state transitions, is organized around four main components: **Workflow**, **BusinessEvent**, **TradeState** and **Instruction**.

Workflow:

A workflow represents a sequence of actions or steps, each one represented by a “workflow step”, that are required to trigger a business event.

The `WorkflowStep` is constructed by the `Create_***WorkflowStep***` functions provided in the model.

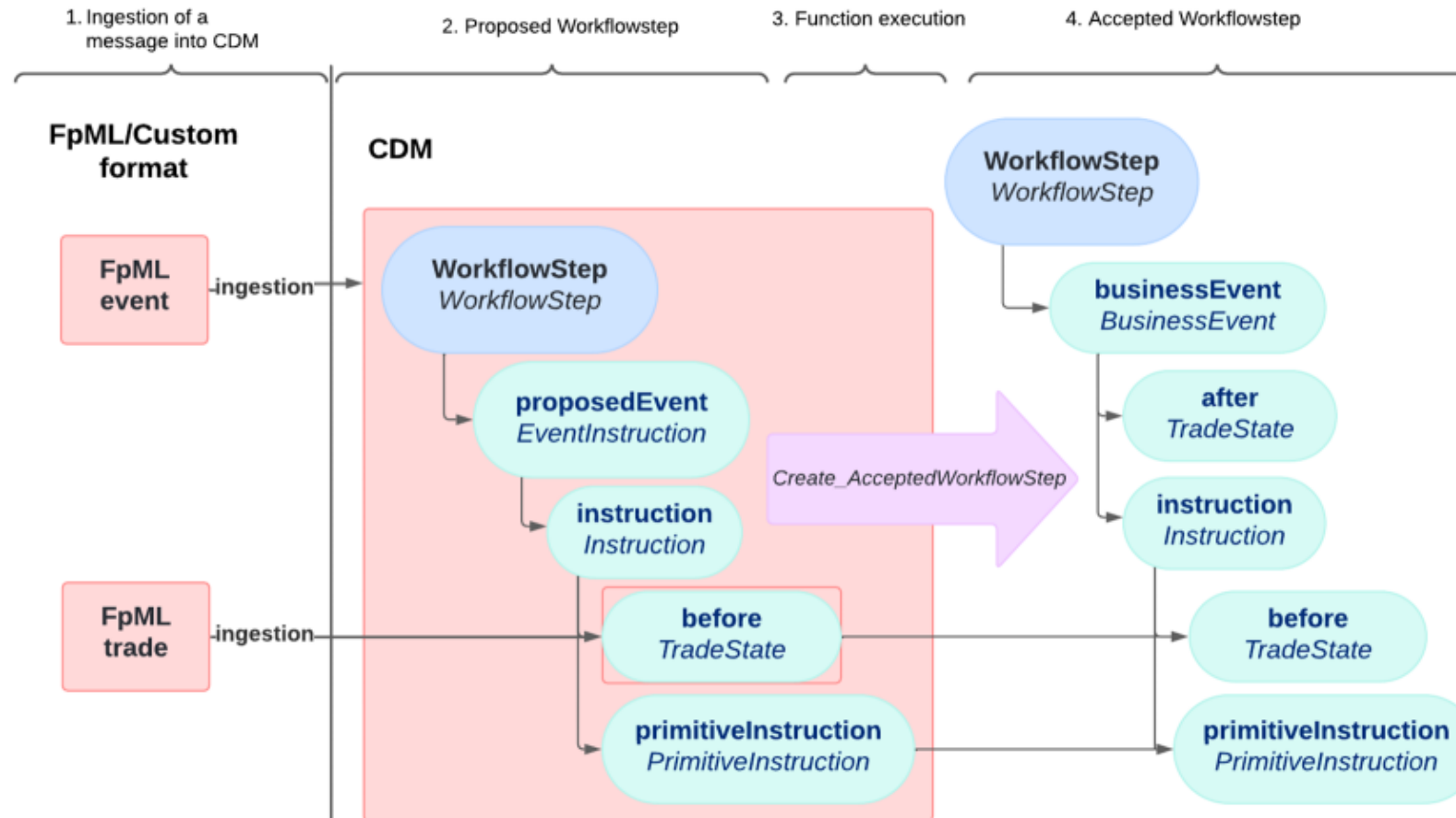
A `WorkflowStep` contains a `BusinessEvent`, Parties, Identifiers, Timestamps and other relevant information.



When a transaction is ingested into CDM a *proposedEvent* is created, containing the instructions to generate the after TradeState.

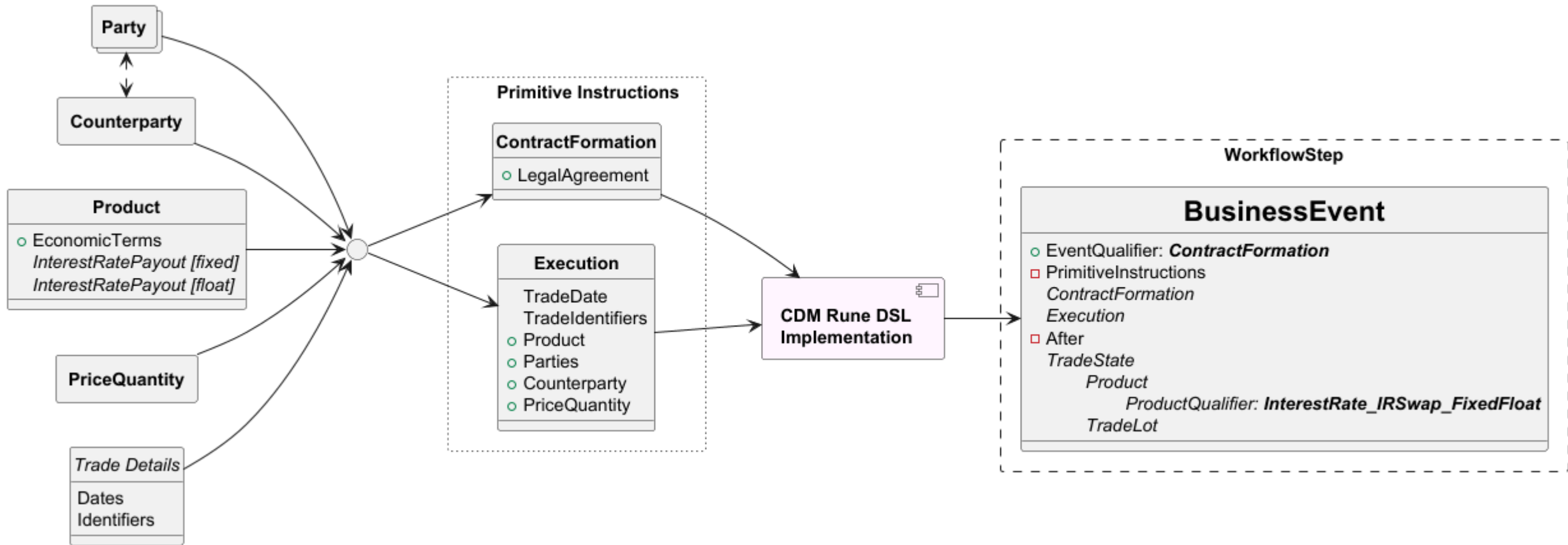
By running the function `Create_BusinessEvent`, the after TradeState is computed and attached to the BusinessEvent together with the instructions and additional information. The `WorkflowStep` is set as accepted and then, the *proposedEvent* is removed.

Construction of CDM Transactions and Lifecycle Events



NOTE: CDM WorkflowStep is included as part of the object that feeds the DRR reporting rules, together with further information specific to reporting.

Construction of CDM Transactions and Lifecycle Events



Construction of CDM Transactions and Lifecycle Events

Events generation

Composition of primitive functions for events creation resulting in a **new trade** and **post-trade** events:

Event Type	Event Name	Primitive operators	Primitive Function Composition (pseudo code)
New Trade	Execution	Execution	Execution(executionInstr) [no before]
	Contract Formation	Contract Formation	If before exists ContractFormation(before) else ContractFormation(Execution (executionInstr))
	Allocation / Partial Allocation	Split Contract Formation Party Change Quantity Change	ContractFormation(PartyChange({ SplitPrimitive(before , qty) }, partyChangeInstr), contractInstr)
	Clearing	Split Contract Formation Party Change	{ ContractFormation(PartyChange(before , partyChangeInstr), contractInstr), ContractFormation(PartyChange(Copy (before), partyChangeInstr), contractInstr) }
Post Trade	Novation / Partial Novation	Split Contract Formation Party Change Quantity Change	ContractFormation(PartyChange({ SplitPrimitive(before , qty) }, partyChangeInstr), contractInstr)
	Give Up	Execution Party Change	

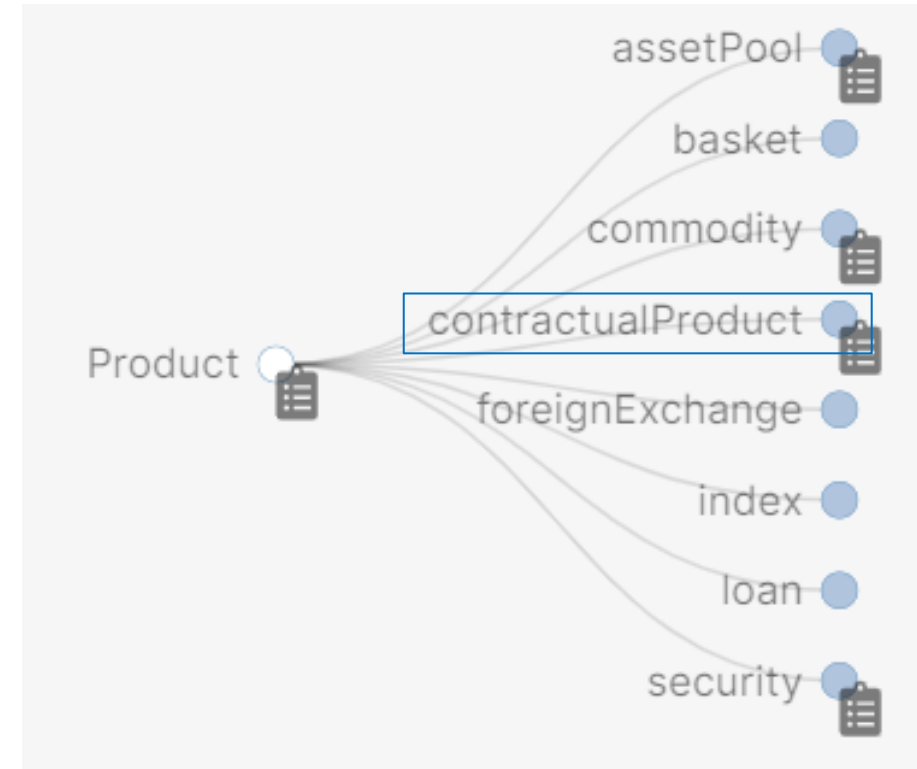
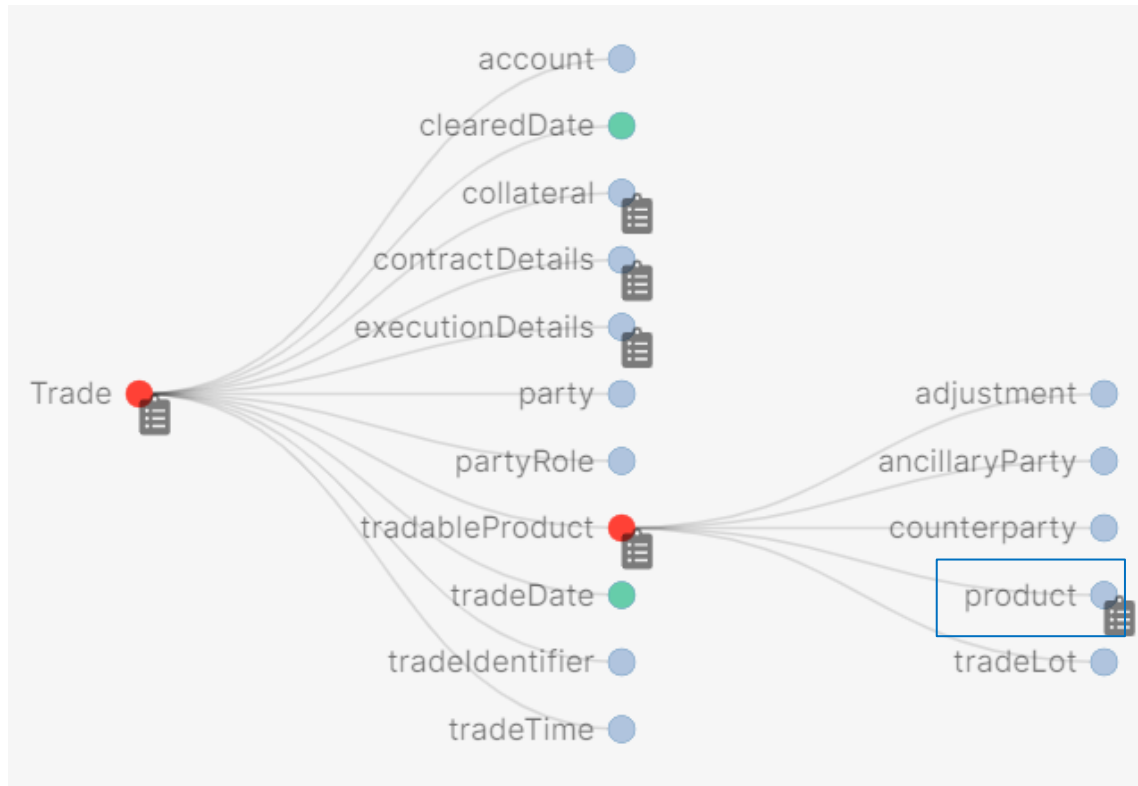
Construction of CDM Transactions and Lifecycle Events

Events generation

Event Type	Event Name	Primitive Operators	Primitive Function Composition (pseudo code)
Post Trade	Increase	Quantity Change	QuantityChange(<i>before</i> , <i>qtyChangeInstr</i>)
	Decrease (Termination)	Quantity Change	QuantityChange(<i>before</i> , <i>qtyChangeInstr</i>)
	Decrease (Partial Termination)	Quantity Change	QuantityChange(<i>before</i> , <i>qtyChangeInstr</i>)
	Renegotiation (other than qty)	TermsChange QuantityChange	
	Compression/Netting	Execution Contract Formation Quantity Change	{ QuantityChange(<i>before</i> , <i>qtyChangeInstr</i>) }, { ContractFormation(Execution (<i>executionInstr</i>)) }
	Reset		
	Transfer		
	Exercise	QuantityChange Transfer	
	Partial Exercise	QuantityChange Transfer	
	Barrier Event		
	Stock Split	QuantityChange TermsChange	

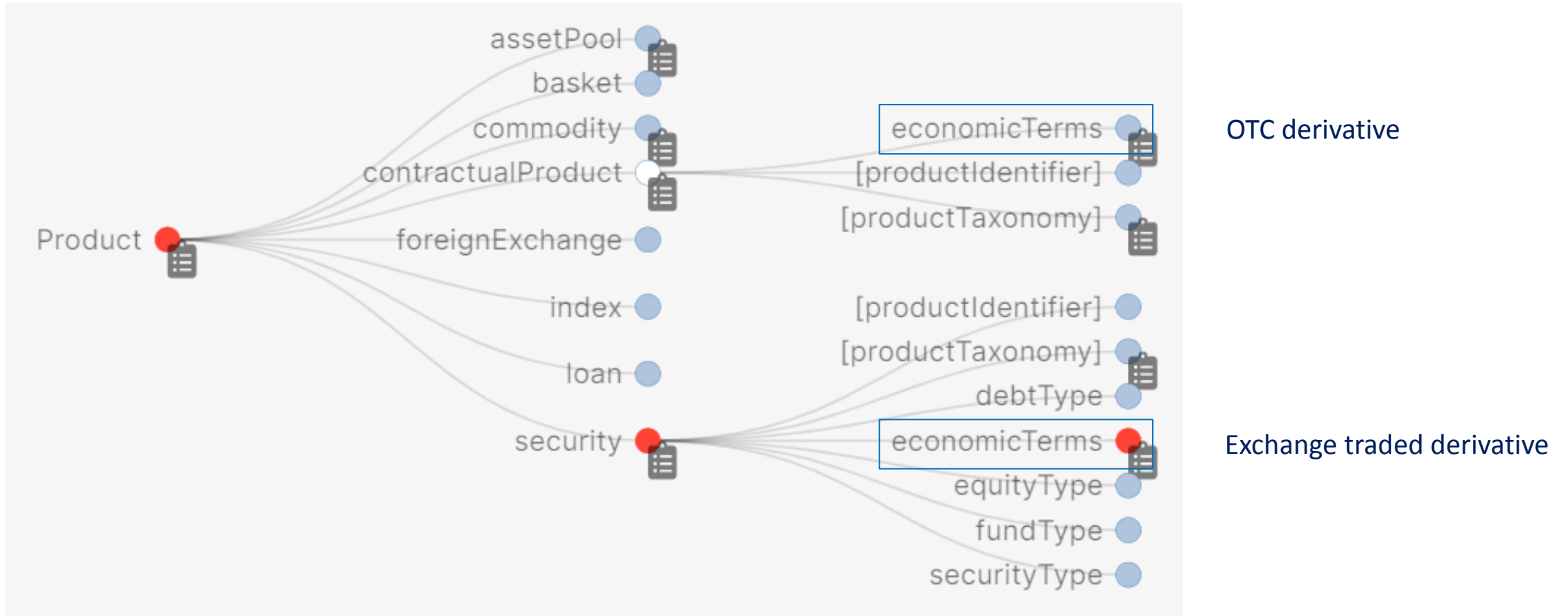
Representation of Products and Core Objects

When representing a business event in CDM, one of the key elements of its structure is the **Trade**. This represents the transaction that has been effected in that event, containing all the financial information and, on top of all, the underlying product.



This product can be a simply executed financial product (index, security, basket, commodity, FX, etc.) or a derivative contract. The latter will be represented through the attribute **contractualProduct** (or as a **security**, if it is an ETD) and it will contain all the information about the legs of the transaction.

Representation of Products and Core Objects



Derivative contracts (such as swaps, options or futures) are represented through a composability principle: several payouts are defined and derivatives are built as a combination of them and their characteristics. Therefore, we are following also another CDM principle: reusability. A payout can be used to build different products from different asset classes.

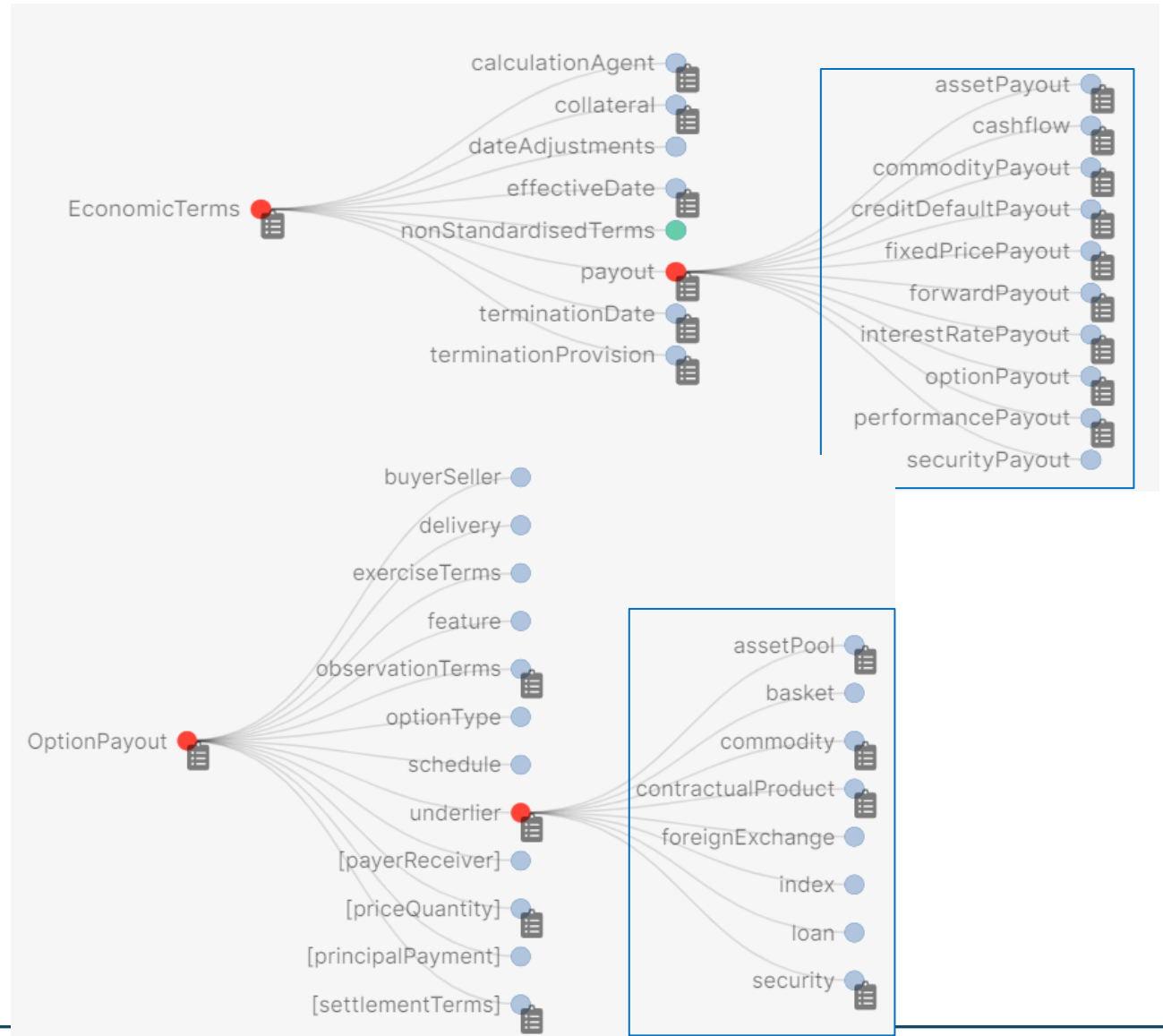
Representation of Products and Core Objects

A few examples of composability:

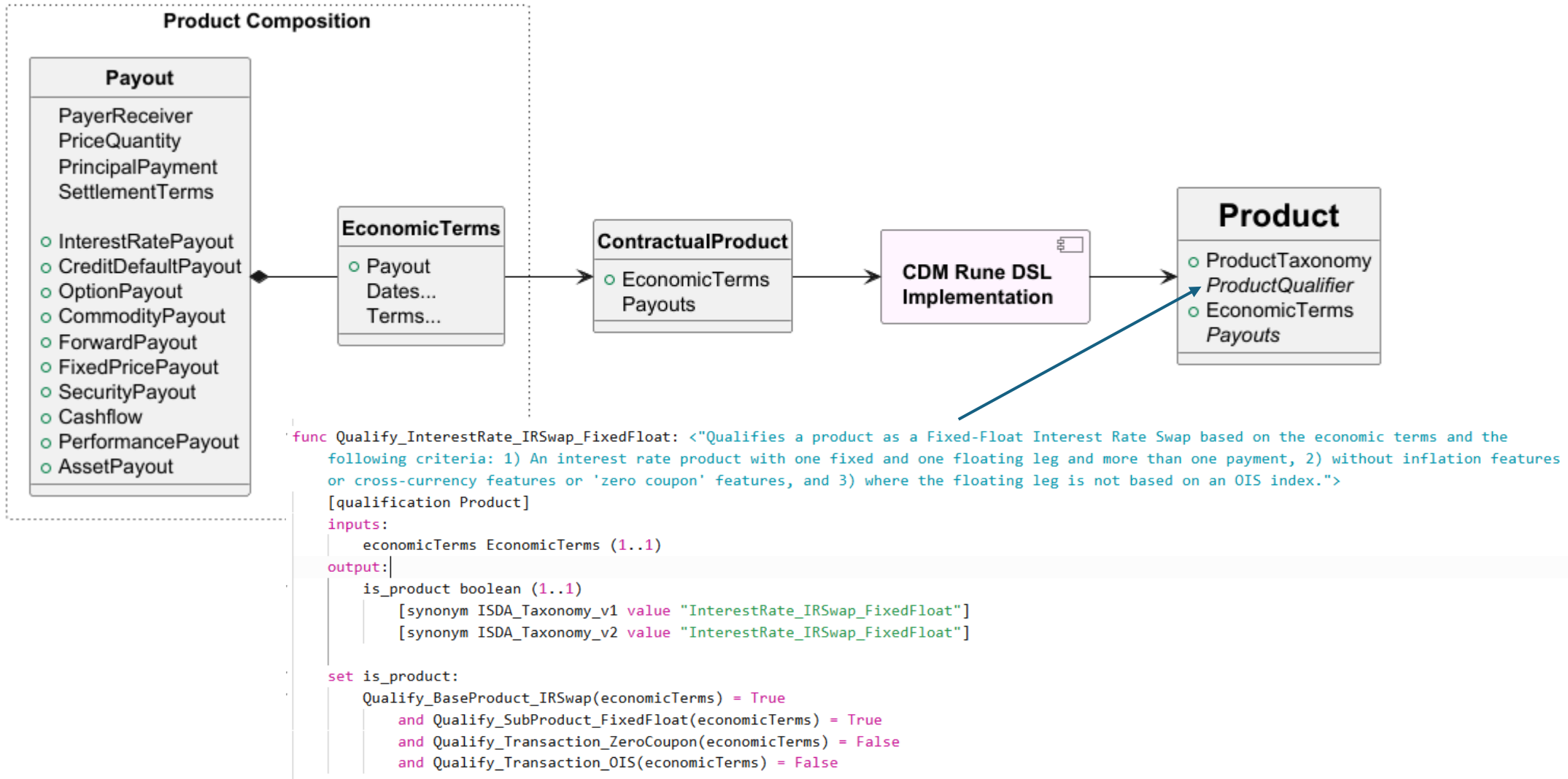
- Credit default swaps:
 - creditDefaultPayout
 - interestRatePayout
- Interest rate swap:
 - interestRatePayout x2
- Commodity option:
 - optionPayout with a commodity underlier

Most of the payouts contain an **underlier** feature that allows to represent the underlying instrument of the derivative. This can be a simple financial product or another derivative (represented identically as shown in the previous slide) if we are representing, for instance, a swaption.

This depiction of products imply that we cannot directly identify them (such as we do in FpML). Consequently, we need to build qualifying functions. These functions identify a product by detecting which payouts build it and the characteristics of those payouts.



Representation of Products - Qualification



How CDM enables Machine-Executable Logic

The Model is exported as a code library, part of the build process involves the generation of source code from the DSL files which is then compiled into binaries.

Where can I get the CDM Artifacts?

- Accessing CDM artifacts from Maven Repository
<https://mvnrepository.com> search **cdm**
CDM artifacts available under the group id **org.finos.cdm**
CDM is available in java, python, kotlin, scala, golang, c#, typescript & others
[cdm-java, cdm-python, *cdm-scala*, *cdm-csharp*, *cdm-daml* ...]
- Building CDM artifact from source code
Download source from GitHub (open-source)
<https://github.com/finos/common-domain-model>
- Rune DSL <https://github.com/finos/rune-dsl>

Latest Versions

Central (365)	
	VERSION ▼
8.0.x	8.0.0-dev.4
	8.0.0-dev.3
	8.0.0-dev.2
	8.0.0-dev.1
7.2.x	7.2.0
7.1.x	7.1.0
7.0.x	7.0.0
	7.0.0-dev.132
	7.0.0-dev.131
	Show 126 more from 7.0.x
6.26.x	6.26.0

Networking Break



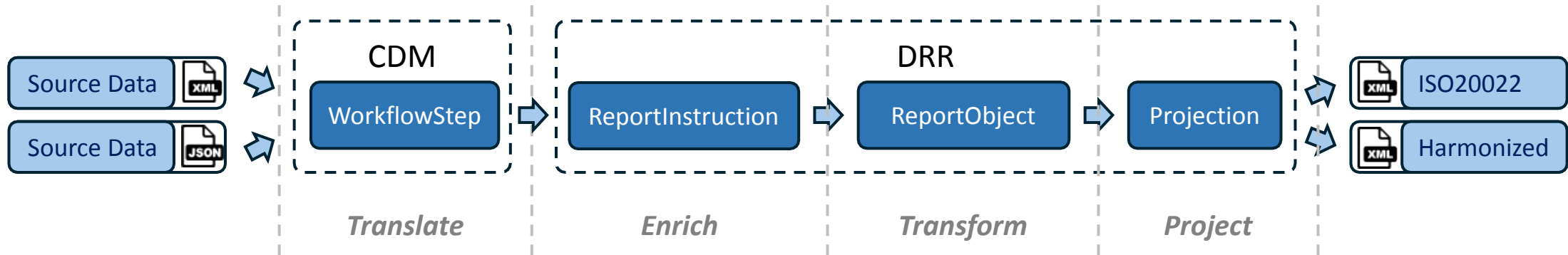
From CDM to DRR- A Technical Walkthrough



How DRR consumes CDM-based transactions and events

DRR Journey

The end-to-end journey from source data to DRR regulatory reports follows four structured steps: Translate, Enrich, Transform, and Projection.

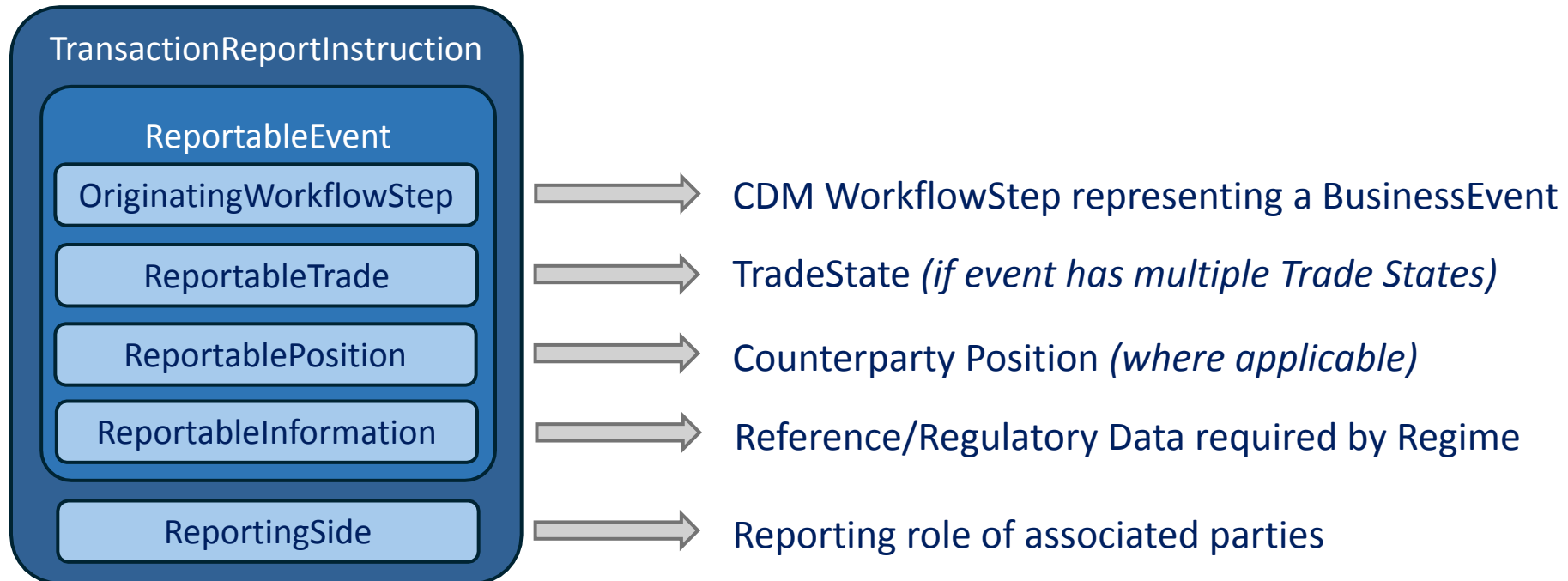


- **Translate:** The step where event information from source data is converted into a CDM representation.
- **Enrich:** The step where a business event is enhanced with regulatory data for the desired reporting Regimes, producing a ReportInstruction used as the basis for reporting.
- **Transform:** The step where a ReportInstruction is transformed into a Report object representing the reportable dataset for a specific regime.
- **Projection:** The step where the reportable dataset is converted into the format required by the regulator or trade repository (TR) for submission.

How DRR consumes CDM-based transactions and events

The Digital Regulatory Reporting is a model that extends CDM, adding regulatory reporting features to it. Therefore, DRR leverages CDM in order to generate and validate reports from business events presented in a CDM format.

In terms of data model, DRR adds a higher layer of attributes that contain all the needed information to generate a well-formed and valid report.

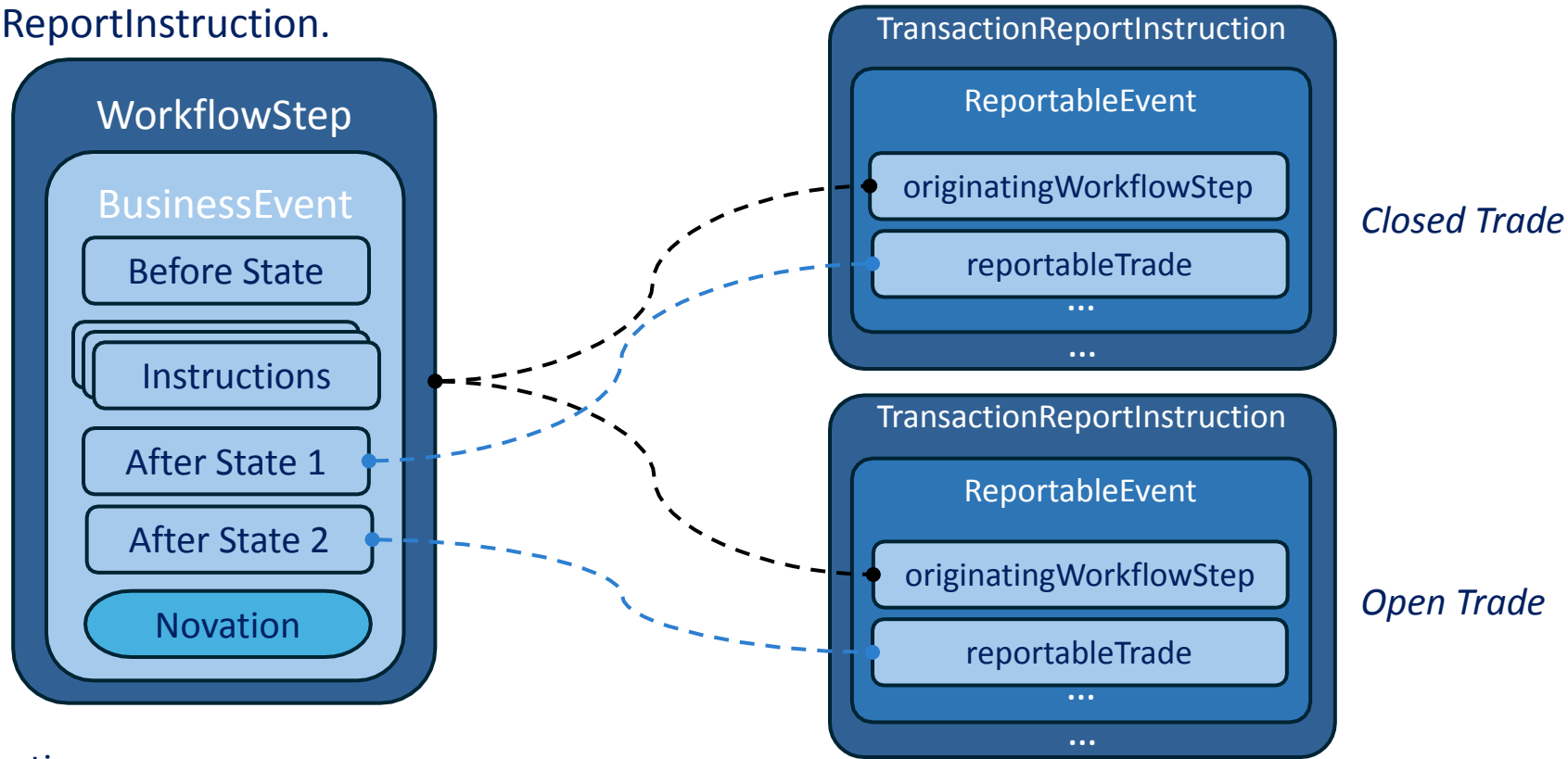


Currently, DRR extends the **ReportableEvent**, creating the **TransactionReportInstruction**. This type is the root element of the DRR and it is from where all reports are generated. This means that the JSON input file of a report generation needs to be a **TransactionReportInstruction**. This type contains a feature needed to specify delegated reporting.

How DRR consumes CDM-based transactions and events

Transaction Report Instruction - Construction

Each Trade State is a unique reportable trade/position; therefore, every state requires a corresponding TransactionReportInstruction.



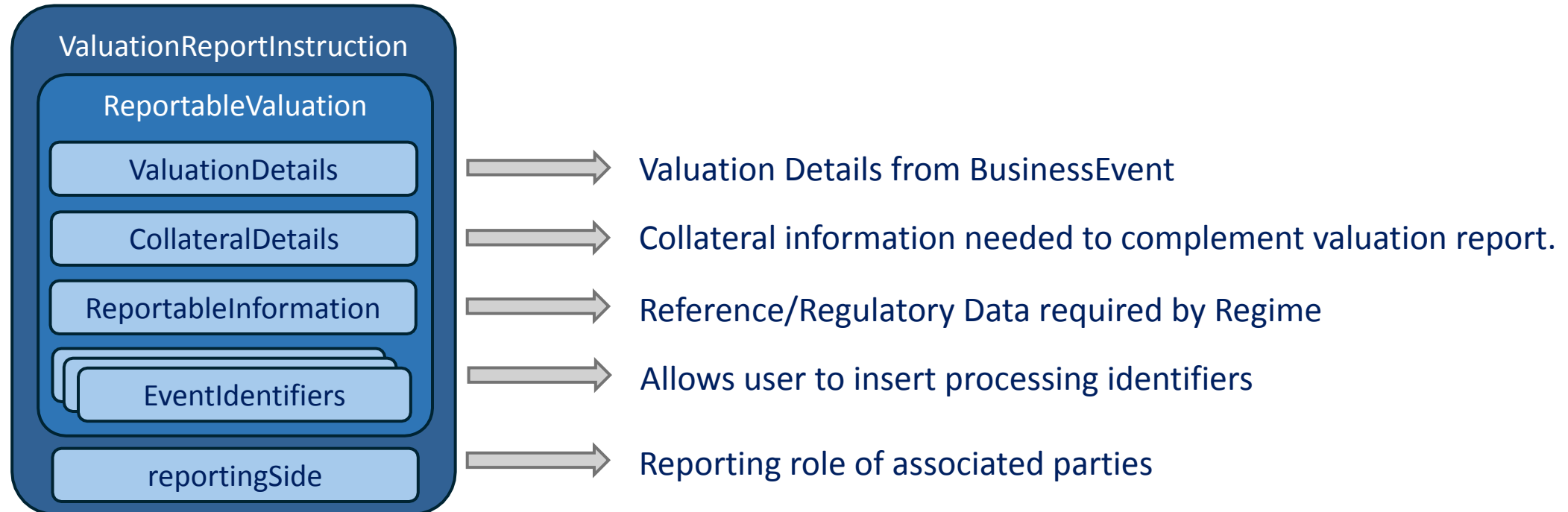
Example: Novation

- Old Trade -> TransactionReportInstruction -> Submission TERM/NOVA
- New Trade -> TransactionReportInstruction -> Submission NEWT/NOVA

How DRR consumes CDM-based transactions and events

Valuation Report Instruction

A ValuationReportInstruction represents valuation data for a trade at a point in time

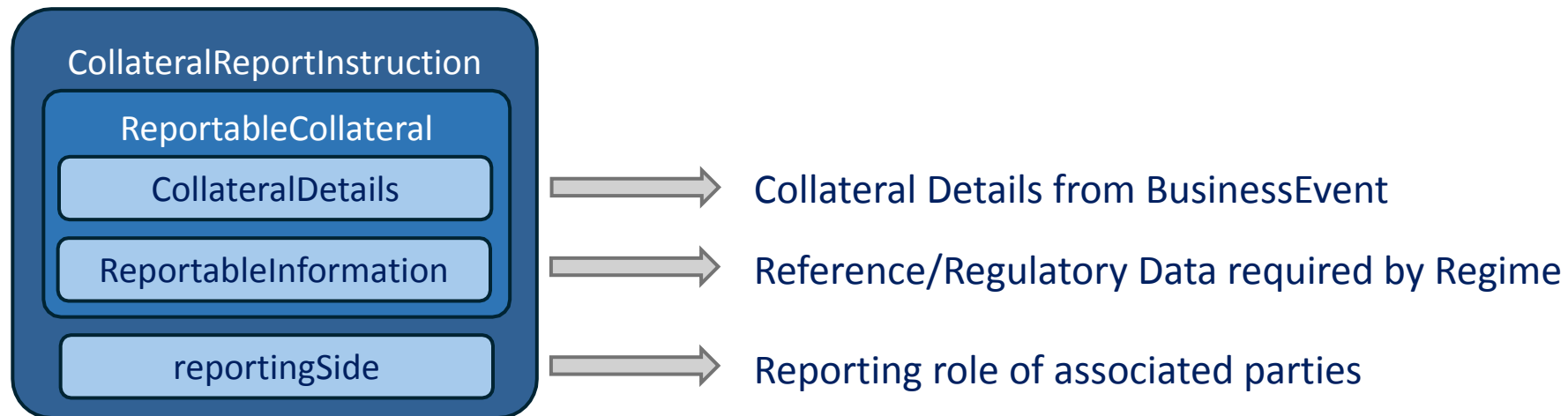


ValuationDetails may be sourced from a Business Event or explicitly constructed to create the Instruction

How DRR consumes CDM-based transactions and events

Collateral Report Instruction

A CollateralReportInstruction (sometimes called margin reporting instruction) represents collateral / margin data associated with trades or portfolios.



CollateralDetails may be sourced from a Business Event or explicitly constructed to create the Instruction

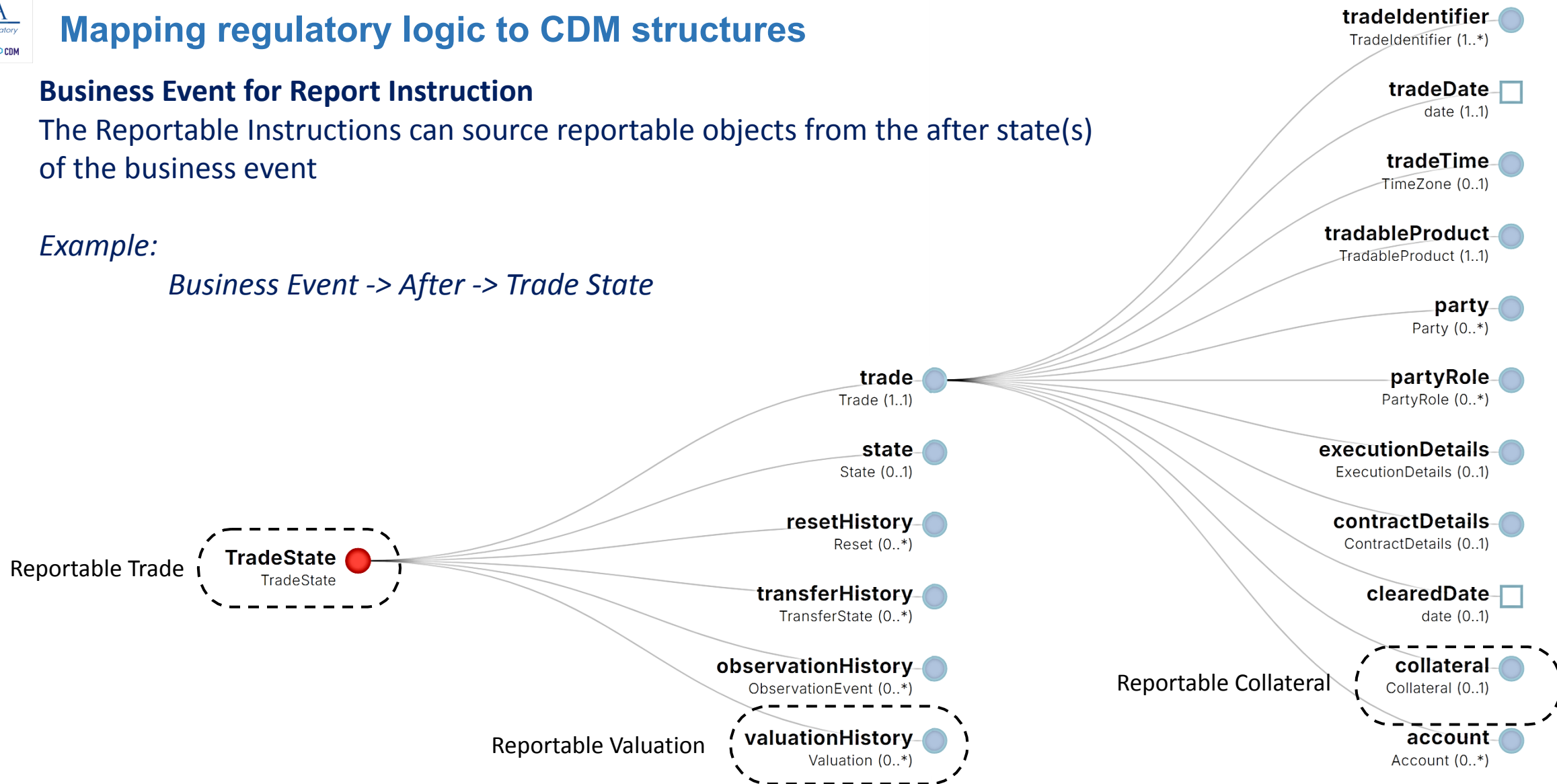
Mapping regulatory logic to CDM structures

Business Event for Report Instruction

The Reportable Instructions can source reportable objects from the after state(s) of the business event

Example:

Business Event -> After -> Trade State



End-to-end walkthrough

Live Demo FSCA ZAEMIR Reports

Technical Q&A and Discussion



Closing and Next Steps



Get Involved- DRR

How to Join Working Groups:

Account Registration

1. click the [Sign Up Now](#) link located in the top right corner of [our homepage](#).
2. Look for an email from isdams@isda.org, which contains instructions for activating your account.
3. If you do not receive the activation email within 1 hour, please contact isdams@isda.org for assistance.

Working Groups Registration

1. Access your [Committee Dashboard](#).
2. Add yourself to the ISDA working groups and distribution lists you wish to join.

For help on anything related to CDM or DRR you can always reach out to us at cdmdrr@isda.org