

ISDA[®]

Safe,
Efficient
Markets



ARK 51

The Common Domain Model: A User Journey from Concept to Production

EXECUTIVE SUMMARY

The Common Domain Model (CDM)¹ is a standardized, machine-readable and executable model for representing financial products, lifecycle events and contractual information. By providing an open-source shared language across firms and platforms, it supports interoperability, automation, resource optimization and reduced operational risk. The CDM is hosted and governed by the Fintech Open Source Foundation² to ensure its continued open-source accessibility.

Although the CDM has seen growing adoption in recent years, there are hurdles to overcome and work involved to integrate it into a system. Each use case will have several approaches, but this paper details how Ark 51³ achieved a successful adoption of the CDM into production for a legal agreement use case.

THE USE CASE

Ark 51 is a document digitization, risk analysis and legal netting consideration application, which uses artificial intelligence to extract data from paper copies of different financial contracts for deeper analysis, categorization, and risk and business optimization. Using a single standard industry digital representation of legal terms would support interoperability and automation, something Ark 51 wanted to help drive rather than observe. The ISDA credit support annex (CSA) was chosen as a first CDM use case.

What makes Ark 51's journey remarkable is the speed with which it went from deciding to implement the CDM to going live. The first use case was identified to map Ark 51's logic to the CDM for one set of variation margin (VM) CSAs. From start to finish, it took only five months to convert from a native Ark 51 data map to a valid CDM representation in production. The firm took a minimum viable product (MVP) approach, ensuring the production value was realized before focusing on more CSA publications, such as initial margin (IM) and legacy documents.

The paper references CDM 6.21.0⁴, which was the production version at the time of writing this paper.

¹ Common Domain Model (CDM), www.isda.org/isda-solutions-infohub/cdm/

² FINOS, www.finos.org/

³ Ark 51, ark-51.com

⁴ Common Domain Model (CDM) version 7.0 is now available

GETTING STARTED

Before adopting the CDM, some fundamental questions must be considered.

- *Is this going to be a re-architecture job or a translation layer?* For example, is the work collateral focused (an area the CDM captures in enough detail to allow for truly native implementation, without external data supplementation)? If the native data representation is populated, a full system migration to native CDM may be possible. More often, however, the maturity of existing systems makes a translation layer the realistic choice, with data either ingested or output in CDM format.
- *If the use case is legal, is the focus the CSA or master agreements?* The CDM is constantly evolving. As it stands, the CDM has comprehensive representation of the most commonly negotiated ISDA CSAs (IM, VM and legacy). However, various master agreements (eg, the ISDA Master Agreement, the Global Master Securities Lending Agreement and the Global Master Repurchase Agreement) are still in development via the CDM community. The underlying requirements of the data that is needed (eg, from a CSA versus a master agreement) will determine whether the CDM can be wholly or partially involved in the digital transformation.
- *What is the use case that could give an MVP?* Ark 51 focused on ensuring a VM CSA could be converted from the native data map to CDM format. If it is possible to find a single flow point from concept to production that can be narrowly defined, like taking a single VM CSA representation, the pipeline can be started and subsequent iterations built upon that pipeline, allowing for a much more seamless transition to CDM adoption. A critical point that some companies miss with CDM adoption is that an all-or-nothing approach is probably not optimal. The CDM covers an enormous range of agreements and events and a full-scale adoption from nothing can quickly lead to much higher onboarding costs. It is recommended to instead adopt in stages and expand when ready and required.
- *Are the right people and resources available for the project?* CDM and digital representation of collateral are primarily a coding challenge, especially translating and mapping from one data representation to another. The team in question will need to understand Rune DSL⁵, an open-source, domain-specific language and the underlying language of the CDM, and be able to write basic data transformations in whichever language is used (for the purpose of this paper, TypeScript is used).
- *Is the native data good enough?* The CDM is represented in heavily nested JavaScript Object Notation (JSON). Everything has an identifying 'key' with an associated 'value'. An example of this is an election in a CSA document, *baseCurrencyTerminationCurrency*, which is the nested key to identify if the base currency of a CSA is also the termination currency, so the 'value' would be a true or false value, referred to as a Boolean type. Understanding the construction is a heavily coded exercise. The other factor to consider is whether the current data map would allow for mapping efficiently. If data is stuck in paper contracts or hosted in a SharePoint, Excel document or Access database, it may be challenging to ensure the requisite data is present and available. There are tools that can help with transforming data from one source to the CDM, as well as companies that have tackled this problem before. While this will have an associated cost, it may be a more economically viable proposal for an initial MVP.

⁵Rune DSL, <https://rune.finos.org/>

MAKING THE MAPPER

Once the initial use case has been identified, the team has been set up and the data has been identified, constructing a mapper from native data structures into CDM format can begin.

Rosetta

The CDM is written in Rune DSL, hosted in GitHub⁶ and distributed as binaries in Maven. Rosetta, which is provided by REGnosys⁷ and helps navigate and interact with the CDM, is one tool that may be used⁸. As the CDM is open source, anyone can view the model. The development team at Ark 51 began with the graphical interface view of the CDM in Rosetta.

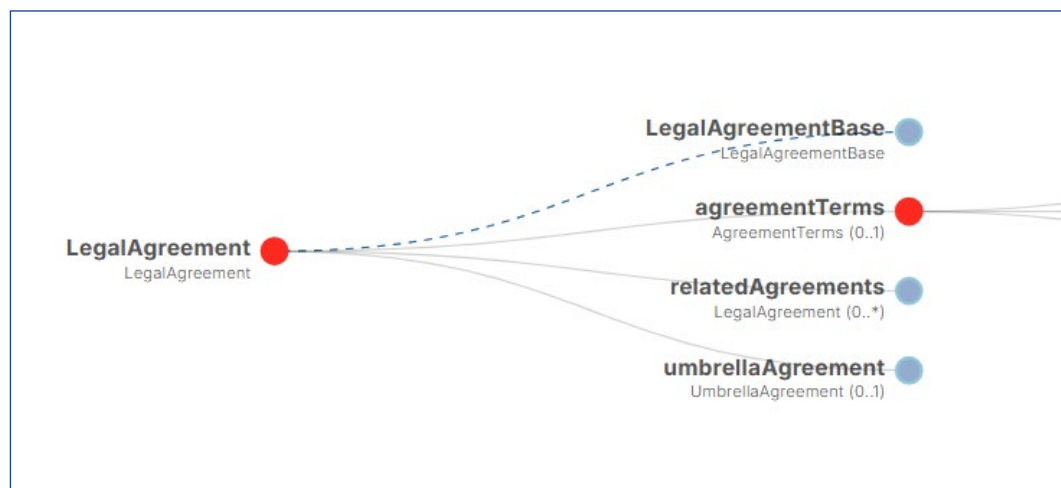
ISDA has also published videos on how to navigate and use Rosetta for the CSA and eligible collateral use cases⁹.

Root Levels

The CDM is, by definition, a way to represent data. This paper does not cover how the data itself is represented in CDM format. As an open-source model, the CDM accepts contributions through its official governance structure¹⁰. The CDM has many 'entry' points, known as root types. These can be considered the outer layer of the JSON object and so don't need to be nested inside anything further.

The screenshots show the flow from the LegalAgreement root level of the CDM to the baseCurrencyTerminationCurrency highlighted earlier (see Figures 1-4).

Figure 1: Beginning from the root level LegalAgreement, move down to agreementTerms



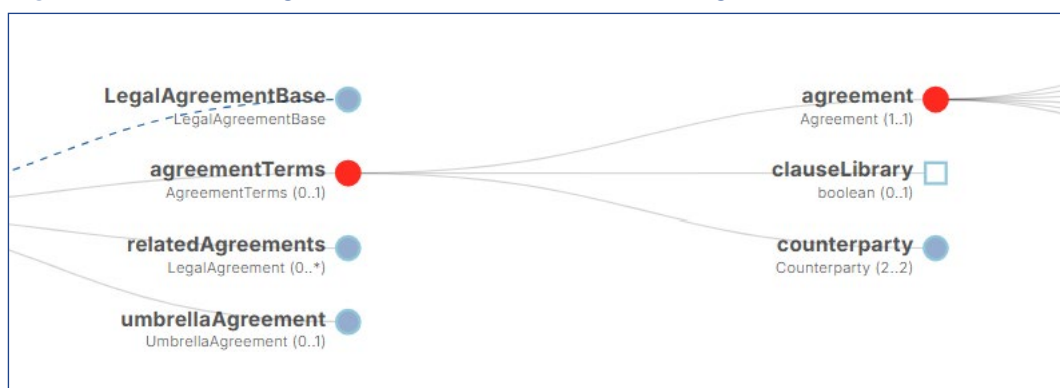
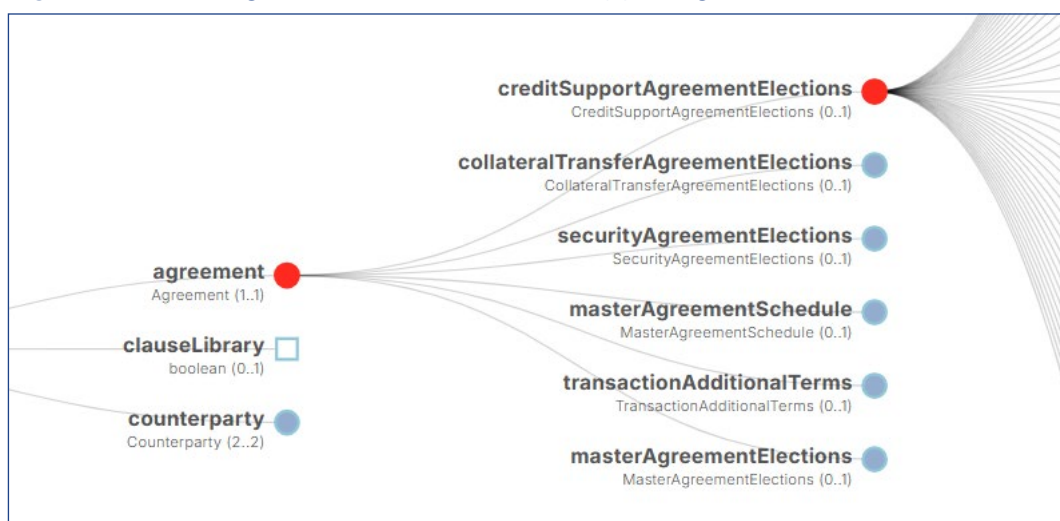
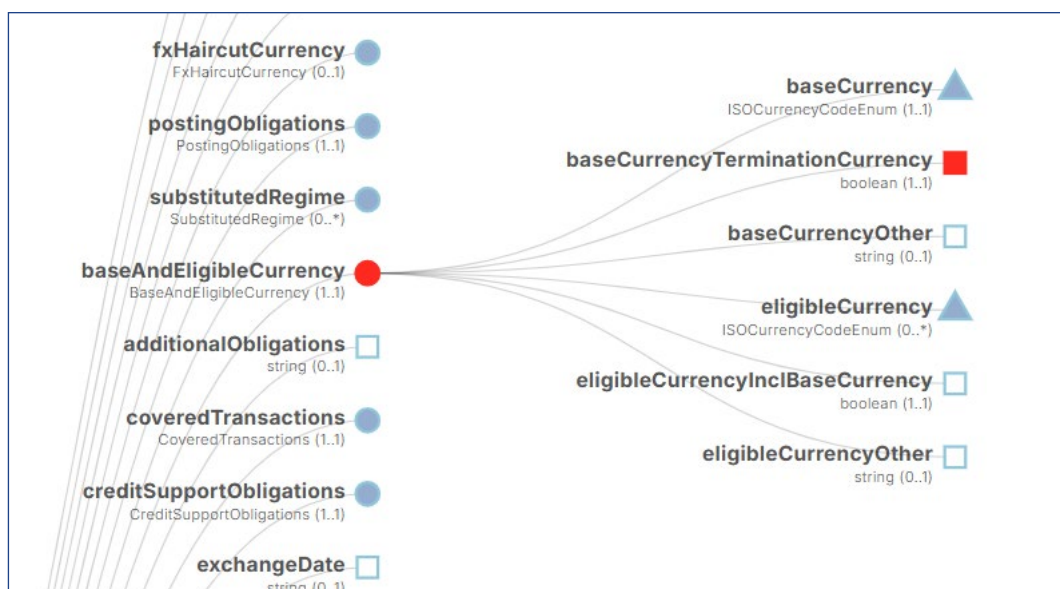
⁶ github.com/finos/common-domain-model

⁷ Rosetta, REGnosys, regnosys.com/solutions/rosetta/

⁸ This paper uses Rosetta screenshots for illustrative purposes and does not represent an official endorsement. Rosetta is free to use on a community license

⁹ CDM Collateral Guidance Video – ISDA CSA Overview Rosetta, www.isda.org/isda-solutions-infohub/collateral-initiatives/cdm-video-csa-overview/; CDM Collateral Guidance Video – Eligible Collateral Overview Rosetta, www.isda.org/isda-solutions-infohub/collateral-initiatives/cdm-video-eligible-collateral-overview/

¹⁰ github.com/finos/common-domain-model/blob/master/CONTRIBUTING.md

Figure 2: From the agreementTerms, move down to agreement**Figure 3:** Inside agreement, move to creditSupportAgreementElections**Figure 4:** Inside the CreditSupportAgreementElections, expand baseAndEligibleCurrency, and locate the baseCurrencyTerminationCurrency

As can be seen in Figures 1-4, an indication of what is present flows through the expanded web-like structure until primitive types (eg, Booleans, strings, numbers or enumerated values) are shown from a list (eg, currency codes).

In Rune DSL, it's important to consider the cardinality notation (0..1) or similar while building the mapper. It is easiest to consider it as (*minimum number allowed..maximum number allowed*).

More explicitly:

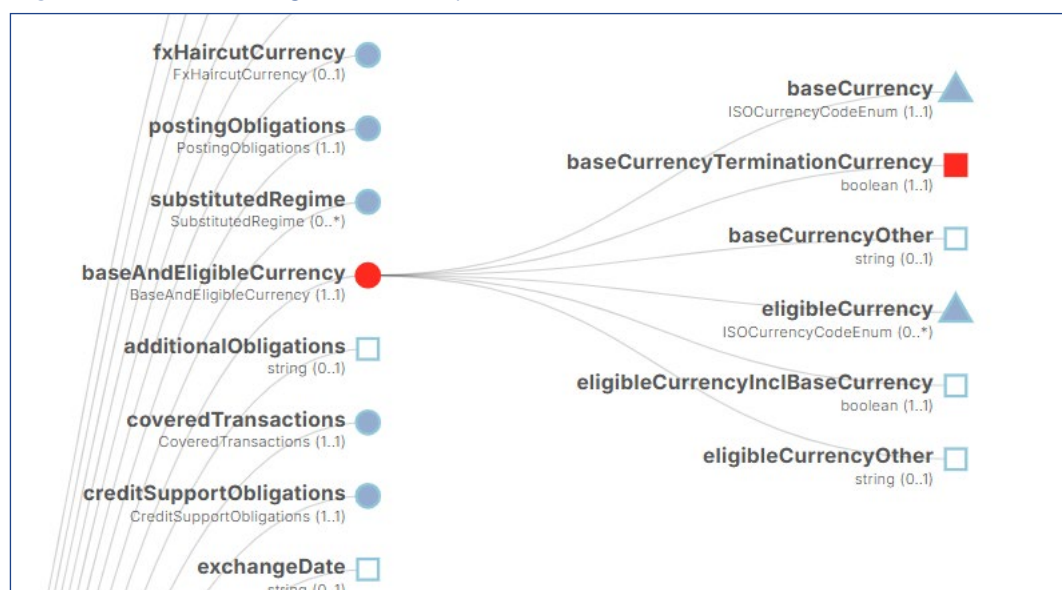
- (0..1) means optional – but if one is provided, it must be only one;
- (1..1) means mandatory – one and only one must be provided;
- (0..2) means optional – one or two could be provided. This would then be represented in an array format;
- (0..*) means optional, and as many as required can be provided; and
- (2..2) means mandatory, and exactly two values must be provided. This example is quite frequent in the CDM when representing legal agreement data, with one value being typically provided for Party A and another value for Party B in the agreement.

Constructing the Mapper, Level by Level

As seen in Figures 1-4, further data representation beyond the root level legal agreement is already quite nested. The most sensible recommendation is to consider at each level what is required from the most detailed level up and build what is needed for each level until the root level is reached.

For example, if a firm wants to represent the fact that the base currency of a contract is the same as the termination currency, information is contained in the *baseAndEligibleCurrency* attribute. More specifically, this is the attribute where information about the base and eligible currency of the legal document is stored (see Figure 5).

Figure 5: baseAndEligibleCurrency attribute



This attribute has six different values. The cardinality notation is used to quickly determine whether it is mandatory and the type required. In the graphical view (see Figure 5), primitive values are denoted by squares, enumerated values by triangles and more complicated types by circles.

Table 1

Name	Type	Mandatory
baseCurrency	ISO Currency Code	Yes
baseCurrencyTerminationCurrency	Boolean	Yes
eligibleCurrencyInclBaseCurrency	Boolean	Yes
eligibleCurrency	ISO Currency code	No
baseCurrencyOther	String	No
eligibleCurrencyOther	String	No

Looking at the different values, the top three rows in Table 1 are mandatory. Information is therefore needed on what the base currency is, if the termination currency is also the base currency and if the eligible currency includes the base currency. In addition, other eligible ISO currency codes are listed and, if the ISO code is not available, then a string fallback option can be provided.

Using typescript notation, the Ark 51 mapper is constructed to represent that same logic (see Figure 6).

Figure 6: A TypeScript representation of the same CDM attribute for `baseAndEligibleCurrency`

```
export interface CDMBaseAndEligibleCurrency {
  baseCurrency: ISOCurrencyCodeEnum | 'Other';
  baseCurrencyOther?: string;
  baseCurrencyTerminationCurrency: boolean;
  eligibleCurrency?: ISOCurrencyCodeEnum[];
  eligibleCurrencyInclBaseCurrency: boolean;
  eligibleCurrencyOther?: string;
}
```

Once the data is retrieved from the in-house system, a valid CDM representation can be made (ie, the base currency and whether it is included as an eligible and termination currency is defined), then a function can be created that returns those values in this small valid snippet of the CDM. A basic example is shown in Figure 7, utilizing USD as the currency code.

Figure 7: A valid CDM representation of the `baseAndEligibleCurrency` type

```
{
  "baseCurrency": "USD",
  "baseCurrencyTerminationCurrency": true,
  "eligibleCurrencyInclBaseCurrency": true
}
```

A more complicated example is provided in Figure 8.

Figure 8: A more complicated valid representation of baseAndEligibleCurrency. This one contains an array of eligible currencies

```
{
  "baseCurrency": "USD",
  "baseCurrencyTerminationCurrency": true,
  "eligibleCurrencyInclBaseCurrency": true,
  "eligibleCurrency": [
    "USD",
    "EUR",
    "GBP",
    "JPY",
    "AUD",
    "CAD",
    "CHF",
    "CNY",
    "SEK",
    "NZD"
  ]
}
```

Now a representation has been obtained of a valid baseAndEligibleCurrency, the next step is to go further up the pipeline. The current full JSON output in CDM is shown in Figure 9.

Figure 9: The 'full' CDM representation capturing the baseCurrencyTerminationCurrency election. This is not currently valid CDM

```
{
  "agreementTerms": {
    "agreement": {
      "creditSupportAgreementElections": {
        "baseAndEligibleCurrency": {
          "baseCurrency": "USD",
          "baseCurrencyTerminationCurrency": true,
          "eligibleCurrencyInclBaseCurrency": true
        }
      }
    }
  }
}
```

The problem that arises is that the mandatory fields inside the *creditSupportAgreementElections* are, by necessity, expansive. At this step, it is required to iterate through all the *creditSupportAgreementElections* and they must be populated and mapped if they are mandatory fields. This process is time-consuming but not overly complicated. At each level, a valid representation needs to be created – or a mapper to establish that valid representation to be more accurate. This is repeated until the top root level is reached. Much of this work is highly repetitive, albeit with different underlying types and representations. It is recommended to follow the principles used by the CDM to reduce repetition by extracting shared logic to be reused for things like extracting a value from a list of the in-house representation or use a general mapper to extract the raw value of each data point and ensure it is assigned the correct key. Even when reusing shared logic, a full CDM mapping for an MVP case can grow very quickly. However, once the MVP case has been completed, subsequent stages will require less engineering, as shared functionality can be reused.

Mapping Lists to Lists

A further area to be explored is the mapping from native code to the CDM. A lot of values in the CDM are enumerated lists, which can be selected from a subset of string values. Many of these are standard lists, such as ISO currency codes, but some of the enumerated lists are non-standard and are designed to capture the most likely outcomes without allowing for excessive variance.

An example of this is the *creditSupportAgreementMarginType*, which has an optional value of a *CollateralMarginTypeEnum*, consisting of either *VariationMargin* or *InitialMargin*. It is highly likely that an internal native system will need to map from one set of values to a value in a list, based on the number of lists that capture detail in the CDM. To accommodate this, there are multiple options available. An option that covers most of the bases is to use an AI model like ChatGPT with an example prompt:

“Here is a list of options [PROVIDE OPTIONS], considering the value provided of ‘United States dollars’ please return the most likely option this would represent”

It should be stressed that this is a potential example that could work rather than a recommendation. In-house teams will need to establish the different risks associated with using AI for list matching.

Keeping on Top of Changes

Throughout the graphical layers of the CDM object, many attributes can be optional. But if they are provided, then they themselves have attributes that are no longer optional. In the case of CSAs, it is perfectly legitimate not to have any *CreditSupportAgreementElections* defined, as it is represented by a *(0..1)* notation. This would significantly reduce the value in the CDM representation. But to provide it, every required field (and their respective required fields, etc) must be mapped.

The CDM is an open-source project and is therefore free to use. The working groups are open to the whole community and contributions are encouraged. There is some cost in terms of resources and time to implement and keep up with the CDM, as changes to the code base are frequent. This is not discernibly more or less intensive than updating an internal data map to change with regulations. Release notes are also compiled regularly to help keep pace with changes made¹¹. It is advised to join any working groups that cover an area of interest within the CDM. If something

¹¹ github.com/finos/common-domain-model/releases

appears to be incorrect when building the mapper, users are encouraged to raise the point as an issue and present the case to the relevant working group. There are also very useful digests that anyone can sign up to¹² that give a quick and easy summary of what has changed in the CDM over a given period of time.

After the Mapping, Was It Worth It?

“Absolutely. It was a steep learning curve to begin with, because of how big the namespaces are when you get into the position of actively trying to generate value from the representation. However, diving into the mapping gave us access to a swath of helpful resources to make the journey easier and, while building the mapper, our team found issues on both sides of the fence. Some areas of the CDM needed to change, which gave us exposure to open-source contributing and the network of companies in the CDM infrastructure, and some areas of our own digital maps were enhanced and modified after seeing how the CDM did it. The benefits far outweigh the challenges, even with the initial effort that needs to be expended.

“We are now actively collaborating with ISDA to get more people involved in the CDM, and we think we have the tools necessary to do so. We don’t consider this a space of competition, but one of collaboration. If you want help getting started, please reach out to ISDA or to me directly – we look forward to supporting more people in adopting the CDM ecosystem.”

Paul Hands, chief technology officer, Ark 51

CALL TO ACTION

This paper sets out an example of how to approach adoption of the CDM for legal agreement collateral workflows and what to consider when taking a use case into production. It gives an indication of time frames, examples of mapping logic and build considerations. All use cases and approaches will be different, but this paper demonstrates the approach taken by one firm to provide an understanding and assist in breaking down the barriers of adopting the CDM.

The opportunity now lies in coordinated adoption. As more firms, vendors and infrastructure providers adopt the CDM, legal agreement data can flow consistently across negotiation, onboarding, collateral, risk, capital, reporting and operations, enabling scalable automation, stronger governance and more effective use of technologies.

The question is no longer whether the CDM is ready for adoption in these areas, but how quickly the market can align around the standards.

RELATED ISDA RESOURCES

CDM-Collateral Initiatives Fact Sheet, www.isda.org/2023/04/12/cdm-collateral-initiatives-fact-sheet

CDM-Collateral Developments: Now is the Time to Implement, April 2026, www.isda.org/a/sJ6iE/CDM-Collateral-Developments-0426.pdf

Expanding Legal Agreement Coverage in the CDM: Umbrella Agreements and Contract Amendments, September 2026, www.isda.org/2026/09/04/expanding-legal-agreement-coverage-in-the-cdm-umbrella-agreements-and-contract-amendments/

CDM Collateral Start-Up Guide:

This guide is designed to help users learn more about the CDM for collateral management and how to implement it. This is targeted towards the business user and is structured as a set of short tutorials, giving the opportunity to interact with the CDM and understand its features.

- **Module A: Introduction to the CDM**, www.isda.org/collateral-initiative-start-up-guide-module-a/
- **Module B: Accessing the CDM**, www.isda.org/collateral-initiative-start-up-guide-module-b/
- **Module C: How the CDM Works**, www.isda.org/collateral-initiative-start-up-guide-module-c/
- **Module D: How the CDM is Used**, www.isda.org/collateral-initiative-start-up-guide-module-d/
- **Module E: Collateral Use Cases**, www.isda.org/collateral-initiative-start-up-guide-module-e/
- **Module F: Advanced Features**, www.isda.org/collateral-initiative-start-up-guide-module-f/

CDM Collateral Guidance Videos:

This series of video clips is based on CDM collateral management use cases and provides guidance on the data structures available in the CDM, covering the various data types and attributes and how to locate them.

- **CDM – ISDA CSA Overview Rosetta**, www.isda.org/isda-solutions-infohub/collateral-initiatives/cdm-video-csa-overview/
- **CDM – Eligible Collateral Overview Rosetta**, www.isda.org/isda-solutions-infohub/collateral-initiatives/cdm-video-eligible-collateral-overview/

Webinars

- **The Common Domain Model – Likezero Transforming ISDA Documentation into Data Intelligence**, www.isda.org/isda-solutions-infohub/collateral-initiatives/webinar-cdm-likezero/
- **Artificial Intelligence and CDM – Ark 51’s Journey to Digitize Documentation**, www.isda.org/isda-solutions-infohub/collateral-initiatives/webinar-ai-and-cdm/

- **Using AI and the CDM for Digitized Documentation**, www.isda.org/isda-solutions-infohub/collateral-initiatives/webinar-using-ai-and-cdm/
- **CDM – Digital Documentation Reducing Onboarding Time and Resources**, www.isda.org/isda-solutions-infohub/collateral-initiatives/webinar-cdm-digital-docs/

CDM Test Sample Data

CDM test sample data covering various ISDA credit support annexes types and eligible collateral schedules: <https://github.com/finos/common-domain-model/tree/master/examples/samples/legal-agreements>.

ABOUT ISDA

Since 1985, ISDA has worked to make the global derivatives markets safer and more efficient. Today, ISDA has over 1,000 member institutions from 79 countries. These members comprise a broad range of derivatives market participants, including corporations, investment managers, government and supranational entities, insurance companies, energy and commodities firms, and international and regional banks. In addition to market participants, members also include key components of the derivatives market infrastructure, such as exchanges, intermediaries, clearing houses and repositories, as well as law firms, accounting firms and other service providers. Information about ISDA and its activities is available on the Association's website: www.isda.org. Follow us on [LinkedIn](#) and [YouTube](#).

ABOUT ARK 51

[Ark 51](#) is an AI native digital data extraction and analysis tool. Using cutting edge AI technologies and in house subject matter expertise, paper copies of complicated legal documents can be ingested and digested to extract the crucial data underpinning the document magnitudes faster, cheaper and more accurately than human data entry. This data can then be used to analyze patterns, consider inherent portfolio risk, make instantaneous decisions about close out netting and collateral enforceability, and feed into digital downstream systems via API export through native or CDM format. Unlocking the known but difficult to extract value from the data in a portfolio of contracts saves companies time, provides insight and has a direct positive impact on revenue and profits. Ark 51 makes extracting this insight easy, at a very competitive price. Follow us on [LinkedIn](#) and [YouTube](#)