Credo Core Conceptual Model (3CM)

Business information modelling in a nutshell.

Considering a rework of your data architecture to improve or streamline internal processes or improve compliance with external regulations such as Basel, IFRS17, Solvency II.

A solid Business Information Model (BIM) is a must for any data-related project you set up, regardless of the specific context: business or ICT, informational or operational.

In a nutshell, a BIM specifies the meta information used to achieve a common understanding of Business Objects, their definition, structure and contents as well as all relationships to other Business Objects. The added value lies in architectural and data quality (reliability, accuracy, completeness, consistency and plausibility) by design.

The Five Commandments of Business Information Modelling are:

1. Business-oriented: the perspective of the model should in the first place be business. Object and object relations should reflect the business reality and hence be intuitive to business users. For example, redundant aggregation. Insurance or credit client as unique combination of policyholders. It is clear that in reality an insurance client does not exist, it is an ICT-contraption. The business object 'Insurance Client' is commonly implemented as a normal party next to real parties (legal or natural persons). This is where the problems begin. None of the standard questions one would commonly ask a party can be implemented on Insurance Client. The Insurance Client cannot have a legal type (natural or legal), cannot have an address, an age, etc. The object itself and its relations is prone to excessive modification. Consider a policyholder being removed from a policy. This event modifies the unique combination of policyholders and by consequence the Insurance Client. Abstractions should be based on the business reality. E.g. introduction of funded and unfunded collateral where funded collateral gives a priority claim on a designated asset where unfunded do not.
2. Conceptually clean: once business-oriented objects have been defined, they still need to accurately reflect the legal, economic, modelling, measurement and reporting reality, leaving no room for interpretation. It is clear that semantics are very important here. For instance, many models will make different objects according to the state the insurance policy or financial instrument: simulation, quotation, servicing, litigation. In reality there is only one contract changing state.
3. Holistic: the BIM should cover the entire core business of the company. If a model is not holistic, the same reality will be modelled differently. For instance, consider interest characteristics of financial instruments. These characteristics should be modelled once, consistent for all these instruments. When calculating cash flows or interest rate risk for the entire balance sheet, we want consistency for all instruments and contracts in scope.
4. Agnostic: the BIM should cover all aspects of the business domain, staying away from any type of implementation. For example, when making a model for the calculation of Interest Rate Risk, it is tempting to remove all objects and characteristics that are not relevant for valuation purposes. Hence only the legs of an interest rate swap will be considered relevant, not the swap itself. Later on, it then turns out that the swap with both its paying and receiving leg is needed under the ISDA to be able to properly calculate Credit Risk ….
5. Specific: from an IT perspective it is possible to create a model that is very generic and flexible, able to manage a variety of business objects and easy to modify and extend. The major drawback from a business perspective is that these models do not offer much added value. The added value will need to be put into the IT system by programs. An example from the financial sector would be the implementation of a purchase price of a bond position. The generic IT model will model a date type and date value (value pair) allowing to implement as many dates

as required. A specific model will have the two relevant dates for a bond purchase: trade and settlement date.

Credo Core Conceptual Model (3CM)

Based on 25 years of experience in banking and insurance, Credo has created its own business information model. It serves as a template for all internal, all our banking and insurance software development, and consulting projects (mindmap, documentation, implementation template and training) both in the operational, informational and modelling environments.

Recently the model itself has also been commercialized as a stand-alone product.

As you may have gathered from our logo, we are firm believers in Component Based Development and Object Orientation. The consistent use of these rigorous ICT methodologies underly the success of the 3CM: it ensures maximum structure, reusability and functional robustness. However, the robust foundations of the 3CM do not prevent it from bringing added value to applications based on a non object-oriented approaches. Coming back to the example of the ‘insurance policy’. Is this one object going through different statuses (simulation, quotation, servicing, litigation, write off) or do we have different objects for each status? While we have a clear preference for the first option, we know that it will not be the preferred option for all of our clients. The 3CM is set up such that it can deal with both approaches.

Why 3CM?

We believe that the 3CM strikes the ideal balance between added value (specific) towards our business domain and flexibility (generic). Flexibility is built into our model from a functional point of view and not from a technical point of view. We use inheritance to model meaningful differences between generic and specific aspects, without the excessive use of types and (recursive) associations.

Most market reference data models are very generic and flexible, but with limited added value. The 3CM is different because it was designed by experts in both IT and the insurance and banking business. It is therefore strongly business-oriented; many classes map one on one to a business object, defined in business language.

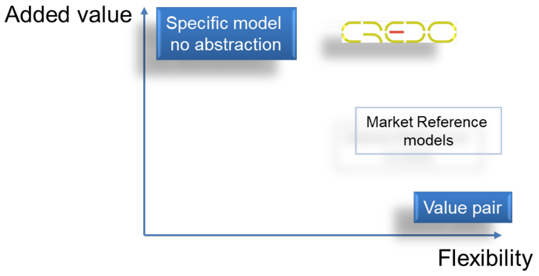
Flexibility is built into our model from a functional point of view and not from a technical point of view. We use inheritance to model meaningful differences between generic and specific aspects without the excessive use of types and (recursive) associations.

The 3CM is managed in Enterprise Architect. The modelling paradigm is UML. Both class and object models are available. The model is versioned. Each release is accompanied by release notes. A number of methodological documents describe the Credo specific modelling assumptions, e.g. how we use enumerators. The language is English.

Want to learn more about business information modelling? Feel free to drop us a line!

Use case

The current version of the model was recently acquired Tier 2 Belgian Bank-Insurer for the implementation of its Company Datawarehouse.


Added value
Specific model no abstraction
CREDO
Market Reference models
Value pair
Flexibility