Ontology for airplane manufacturing

OMG Europe Information Day 2025
Standards for Industrial Systems

COMMERCIAL AIRCRAFT

Michel SAUVAGE, 1ZPA, Airbus SAS

May 15, 2025 | SLB R&D Center, Clamart

AIRBUS

Disclaimer

This document and all information contained herein is the **sole property of Airbus**.

No intellectual property rights are granted by the delivery of this document or the disclosure of its content.

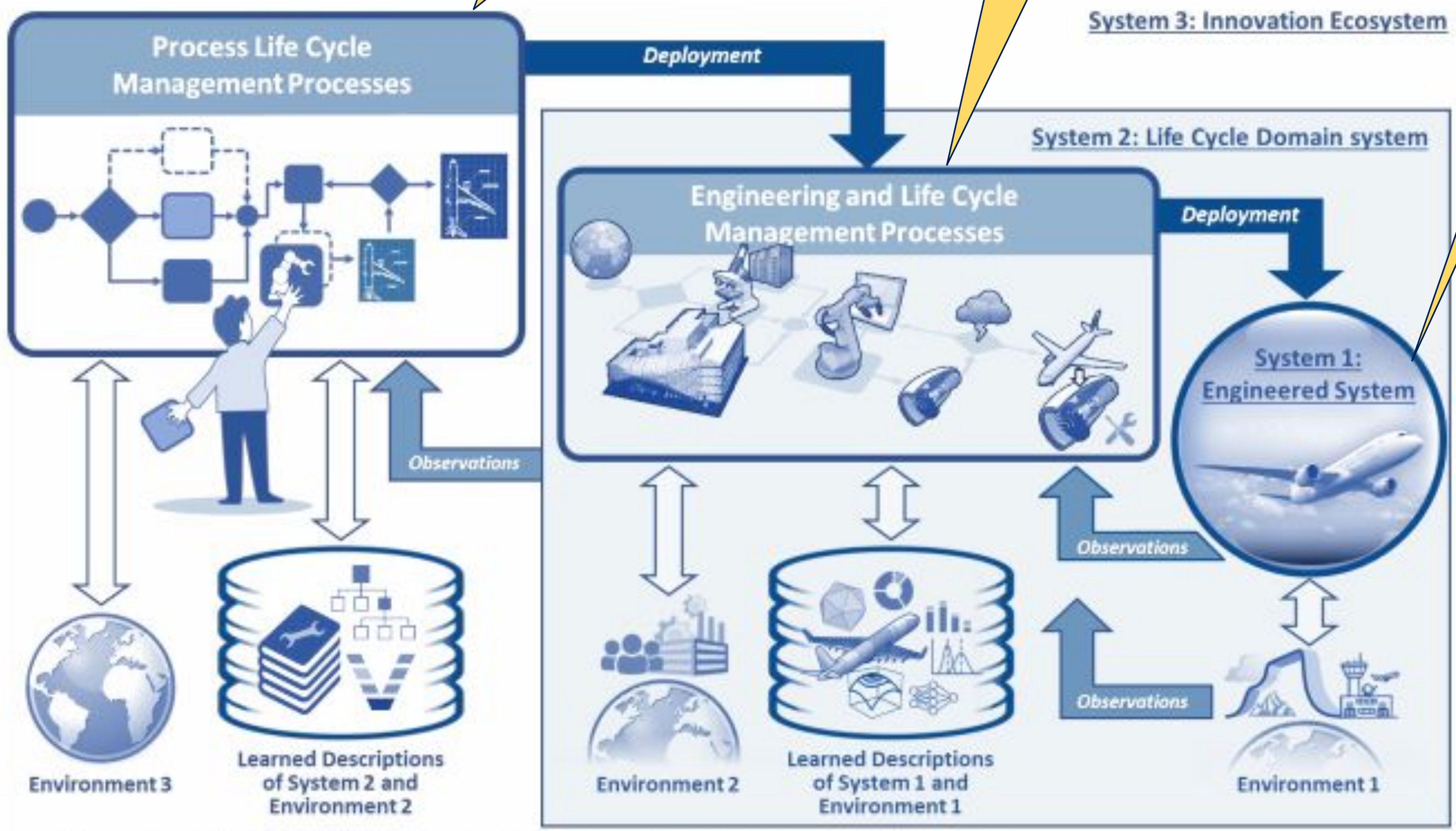
This document **shall not be reproduced or disclosed** to a third party without the expressed written consent of Airbus.

This document and its content **shall not be used** for any purpose other than that for which it is supplied.

Request from the field - voice of practitioners

- Data volume is **increasing**, regulations are **tightening**, and data is now considered a **critical business asset**. However, the data is **partitioned**, lacks business **semantics**, and must be **exchanged** across numerous applications and partners.
- Existing data representation standards are **fragmented** (OWL, BFO, E/R, …) overly focused on **storage** representation rather than **semantics**, and not really **actionable**.
- This results in **low data value**, poor **trust**, limited **reuse**, and inefficiencies that require exponentially **more effort** for management and interoperability, all while regulatory **risks are increasing**.
- A **new standard** is needed to define **actionable semantics** as a foundational mechanism for **data interoperability**.
- **Engage the community** to develop a clear **specification** for this new standard.
- Establish a **solid foundation** for next-generation **Digital Engineering Platform** that are robust, scalable-by-design, and compliant-by-design.

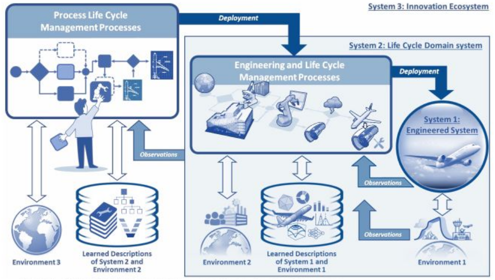
Our reality



From the **system to be built**, to the **system that develop**, to the **system that learns (self improving)** require seamless information flow

INCOSE ASELCM Pattern

AIAA Digital Engineering Integration Committee

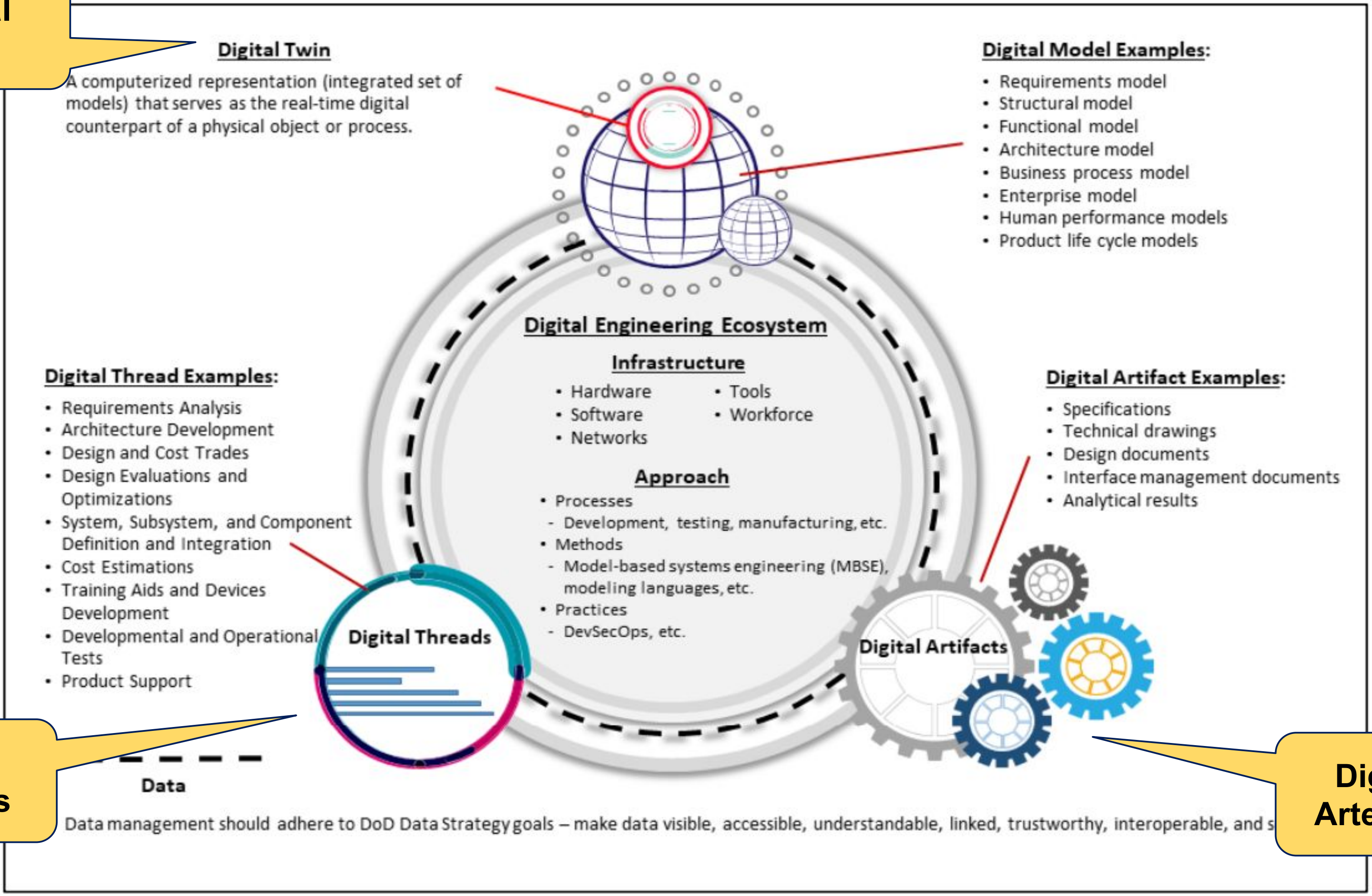


Airbus Amber

Digital Engineering Ecosystem

DOD INSTRUCTION 5000.97 - DIGITAL ENGINEERING

Digital Twin



Digital Threads

Digital Artefacts

→ *Require a flexible framework, **founded on Ontology**, for seamlessly integrating technical data and knowledge across a system's life cycle. It connects trusted data, manages digital models, and transforms data into actionable information for decision-making and feedback.*

Ontology benefits targeted

Mastering Complexity: Digital engineering systems are increasingly complex. An ontology offers a way to structure and organize this complexity by clearly defining concepts, their relationships, and their behaviors, thereby facilitating system design, analysis, and maintenance.

→ To represent the reality

Clarification and Standardization of Meaning: An ontology provides a shared vocabulary and a formal structure to define concepts, their properties, and their relationships. This eliminates ambiguity and ensures a consistent interpretation of data across the various tools and systems that comprise the digital thread. Without an ontological foundation, the same terms can have different meanings in different contexts, hindering data integration and analysis.

→ To agree on the understanding

Semantic Interoperability: Ontology acts as a semantic bridge between heterogeneous systems that use different data models. By aligning these models with a common ontology, it facilitates the exchange and integration of information in a meaningful way, beyond simple technical compatibility. This allows for the combination of data from diverse sources and their holistic analysis.

→ To connect to data

Foundation for Artificial Intelligence and Machine Learning: Ontologies provide a structured knowledge base that can be leveraged by AI and machine learning techniques. They can assist in the extraction of relevant features, the interpretation of results, and the improvement of the algorithms' understanding of the domain.

→ To operate

→ requires a formal ontology language

Ontology language key expectations

Obviously

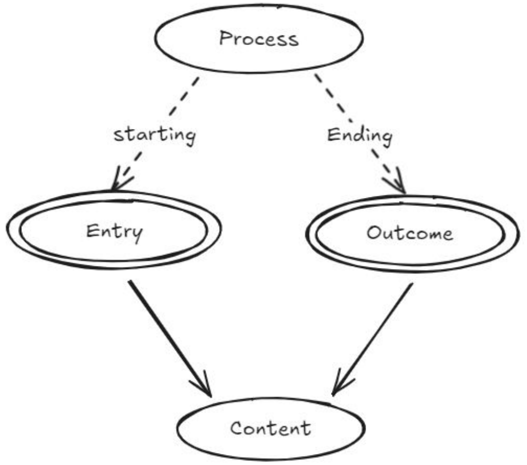
- Syntax to avoid confusion
- Formal semantic
- **Concept** as the way to represent part of reality vs **data** as the way to store concept
- Terminology and community
- Classification (instance, sub type)
- Expressiveness - see after (Time, State, Event, …)

But also

- Modularity - packaging vs composability
- Boundary and connector - clarity comes with contours
- Ability to structure Universe of Discourse - Governance
- Flexible Classification - Uncertainty requires flexibility
- Property dimensions - ISO 80000 (*ilities* - https://en.wikipedia.org/wiki/List_of_system_quality_attributes)

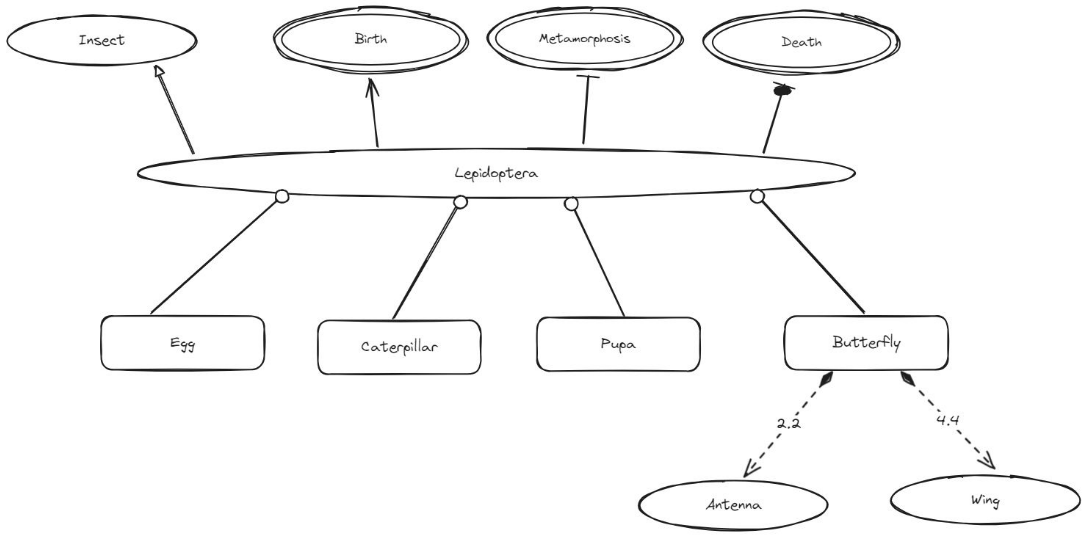
Integratable with Data modeling language (E/R) - see after

Expressiveness at the service of practitioners



Our entity of interest - Product/Program/Enterprise Ecosystems - is by nature complexe and dynamic that requires a “rich” language (about 50 artefacts vs 150 for BPMN e.g.):

- **Period** - to slice the time
- **Rule** - to define functional dependencies (physical laws e.g.)
- **State** & **Event** - to depict change
- **View** - to specify synthesis



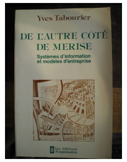
14.5 Problems of terms and data formalizations

It is of course necessary to give oneself a code of good conduct to take into account, as best as possible, the temporal aspects. This should not make us forget that some fundamental questions remain:

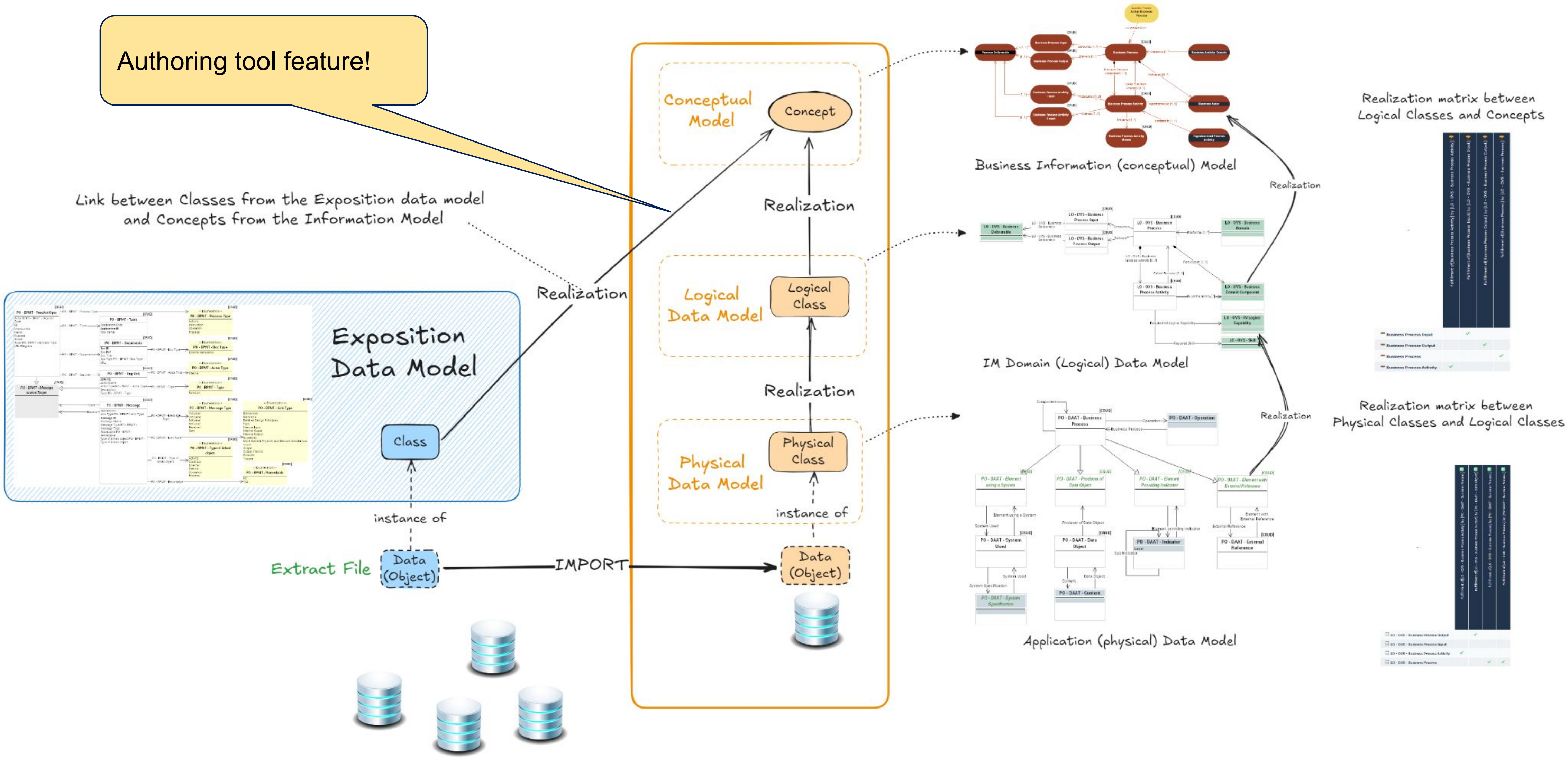
- is it good, or not, to have in the formalism the same notion of entity to represent events and periods?
- is it good to have the entity notation to represent events and durable objects? Durable objects (which are born, live and die) and periods (pure division of time by the pilot)?
- is it good, or not, to have the entity notation to represent objects and past encounters?
- is it good, or not, to have the same notion of relationship to represent ongoing encounters of durable objects, to link objects to the events that concern them, to link events to the periods in which they take place, etc.?

The answer to these questions is not simple, because a certain ambiguity allows flexibility, while clarity brings rigidity. In addition, new dualities will appear in the following chapters, which will provide further fuel to the mill of the supporters of a certain ambiguity. For the time being, I am rather on their side.

1986 - L’autre côté de Merise



Ontology to Data model - Realization between conceptualization levels



AIRBUS

Strong community

Increased dissemination and awareness: An active and engaged community acts as a powerful information relay. Its members share their knowledge, experiences, and the benefits of the new standard, thus reaching a wider audience and accelerating its dissemination.

Collaborative validation and improvement: Through feedback and contributions, the community enables the testing, validation, and improvement of the standard. This collective intelligence ensures a better alignment with real needs and strengthens the quality of the standard.

Mutual support and assistance: Adopting a new standard can be complex. A community provides a space for exchange, problem-solving, and mutual support, thereby facilitating adoption and reducing obstacles.

Creation of resources and best practices: The community contributes to the creation of tutorials, documentation, tools, and best practices around the standard. These resources facilitate its learning and integration into projects.

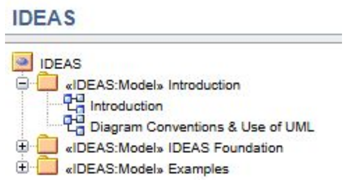
Legitimization and collective buy-in: Mass adoption by the community lends legitimacy to the standard. The more actors who use it, the more it becomes a reference, encouraging others to follow suit and ensuring greater long-term viability.

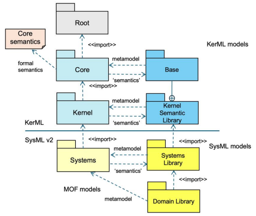
→ No adoption without community

Why we need a new standard… best of breed and fill the gap

We can observe some concerns that have prevented the emergence of an end to end semantic framework

- Lack of aggregate entity and directed connectors (input vs output, begin vs end) → **UML** and **KerML**
- Lack of temporal foundations → **4D approach**, **ISO 19926**, **KerML**
- Incomplete set of semantic relationships (specialization, composition, multi level classification) → **ML2**
- Lack of formalization of abstraction level (conceptualization levels, systemic levels) → **SEAF**
- Lack of clear distinction between entities and theirs properties → **IDEAS Group**, **Ackoff**
- Lack of formalization of family of concepts → **Set theory**, bounded context (**DDD**)
- …





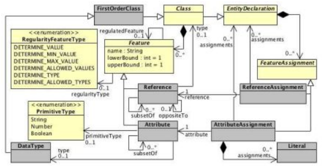
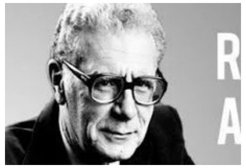


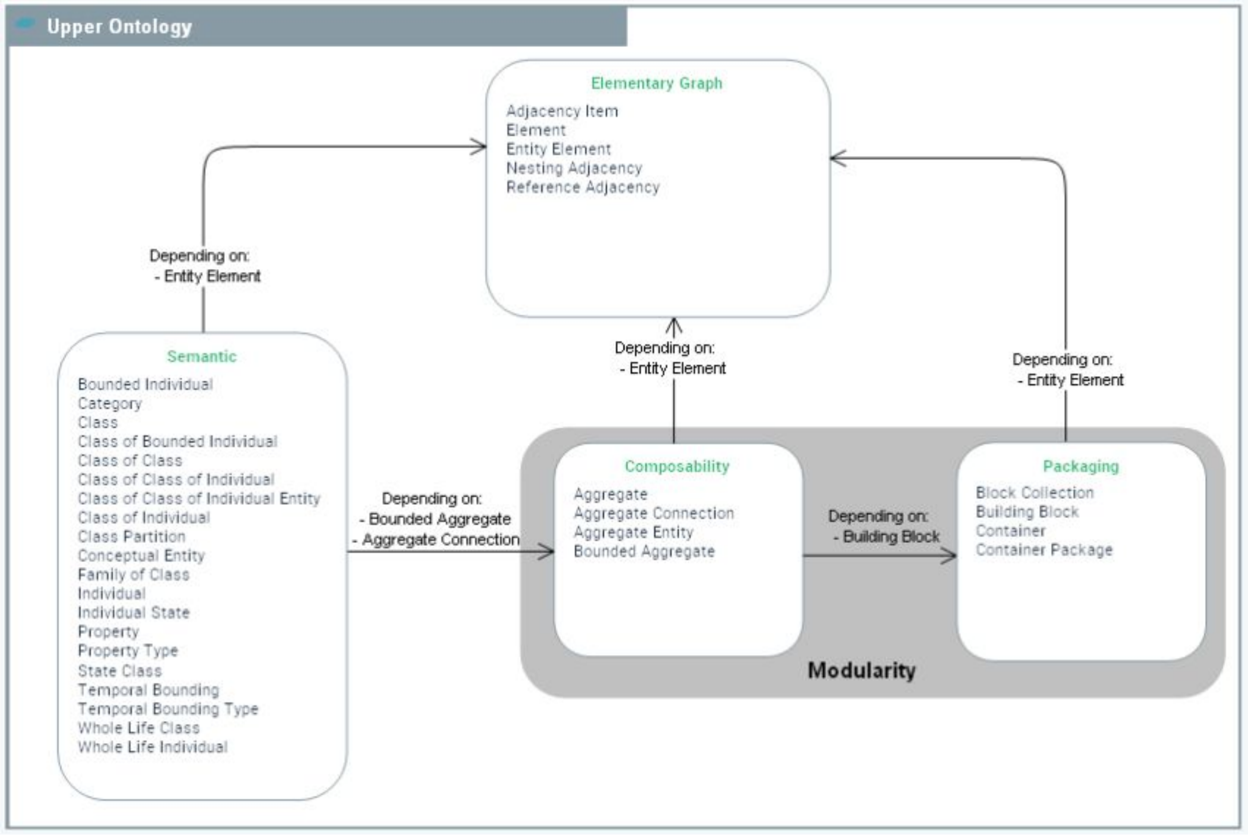
Fig. 8. Features and assignments in ML2.



Very promising work in progress

Proposed by an **OMG member** there is an Upper Ontology domain encompasses the four fundamental domains that establish the foundation for the modeling syntax:

1. **Elementary Graph** defines basic constructs for entities and relationships.
2. **Composability** defines the syntactic constructs used to build Entity Elements that have an internal structure and boundaries.
3. **Packaging** defines syntactic constructs used to organize models in modules called Containers.
4. **Semantic** defines Bounded Individuals (entities that exists over space and time how they are composed and classified, enabling effective representation of meaning.



[Upper Ontology link]

AIRBUS

Next steps

- Calling ontology experts and practitioners, contact us to share your concerns and to collaborate Ontology.Standard@airbus.com
- Visit Upper Ontology (SEAF) site and share your comments
- Identify a solution partner to propose an enabler to test ideas - anticipate end user adoption
- Define the standard strategy to “quickly” make it happen

Thank you

AIRBUS