University of Applied Sciences and Arts Northwestern Switzerland
School of Business

UNICAM Università di Camerino 1336
Università di Camerino

The Convergence of Knowledge Graphs/Ontologies with Enterprise Modelling

Knowledge Engineering SS24

MSc Computer Science

Camerino, 20/05/2024

Dr. Emanuele Laurenzi

Abstraction and Human Mind

–The human mind continuously re-works reality by applying cognitive processes.

– **Abstraction:** capability of finding the commonality in many different observations:

  – generalize specific features of real objects (generalization),
  – classify the objects into coherent clusters (classification),
  – aggregate objects into more complex ones (aggregation).

– **Model:** a simplified or partial representation of reality, defined in order to accomplish a task or to reach an agreement

Dealing with complexity and changes

- If the object you want to create or change is simple, and it is not likely to change, then you can do it directly.

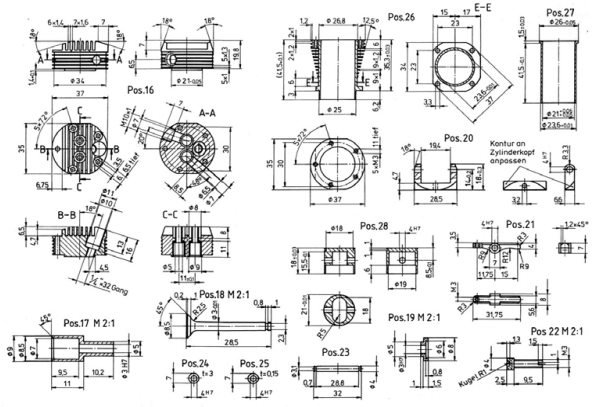
- But…
  - if the object is complex, you can't see it in its entirety at one time and…
  - it is likely to change over time, you need a model.

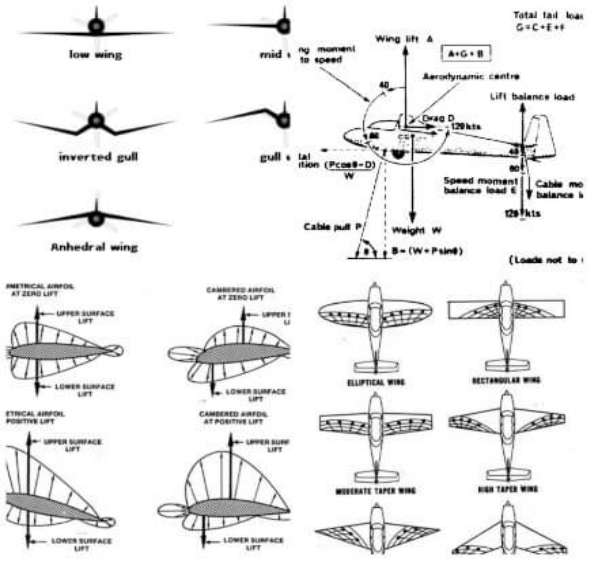
(John Zachmann, 2012)

Through models, we are able to understand and deal with complex systems and their changes.

Complex systems

- Buildings
- Software systems
- Enterprises
- Machines
- Engines
- Electrical networks





Conceptual Models

- "In computer science people use symbolic models to represent their assumptions about a certain domain. These are termed conceptual models" (Guizzardi & Guarino 2023).

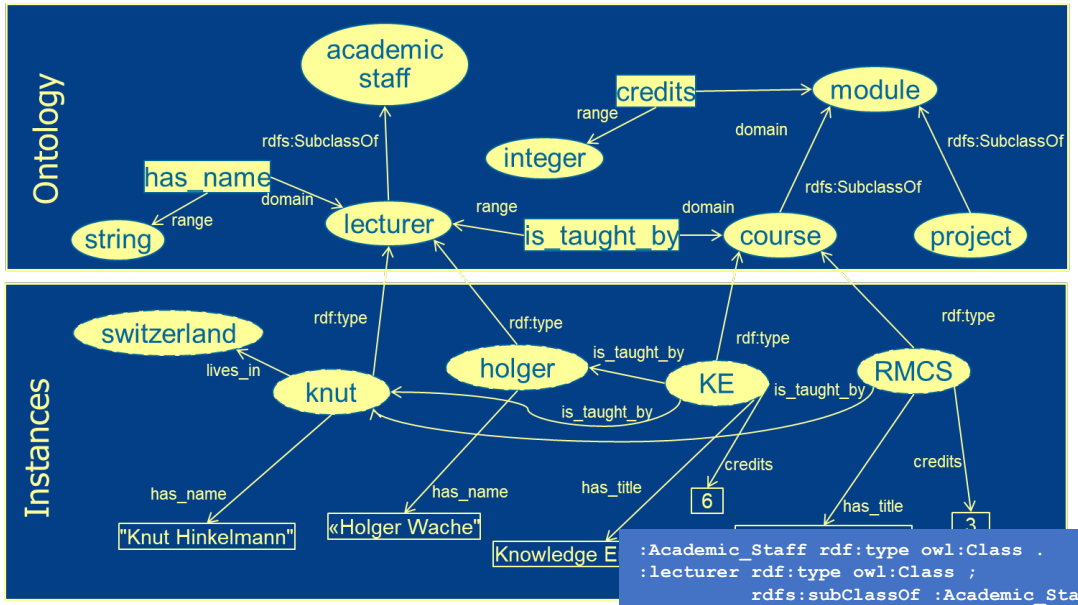
- Conceptual Models are:
  - "created by a (re-)constructing act of abstraction of concepts of a domain which are deemed relevant for a particular purpose" (Frank et al., 2014).
  - "a semplication of a system built with an intended goal in mind" (Bézevin and Gerbé, 2001).

Conceptual Modelling = Knowledge Engineering

– A knowledge base is a representation of reality.

Reality

Model

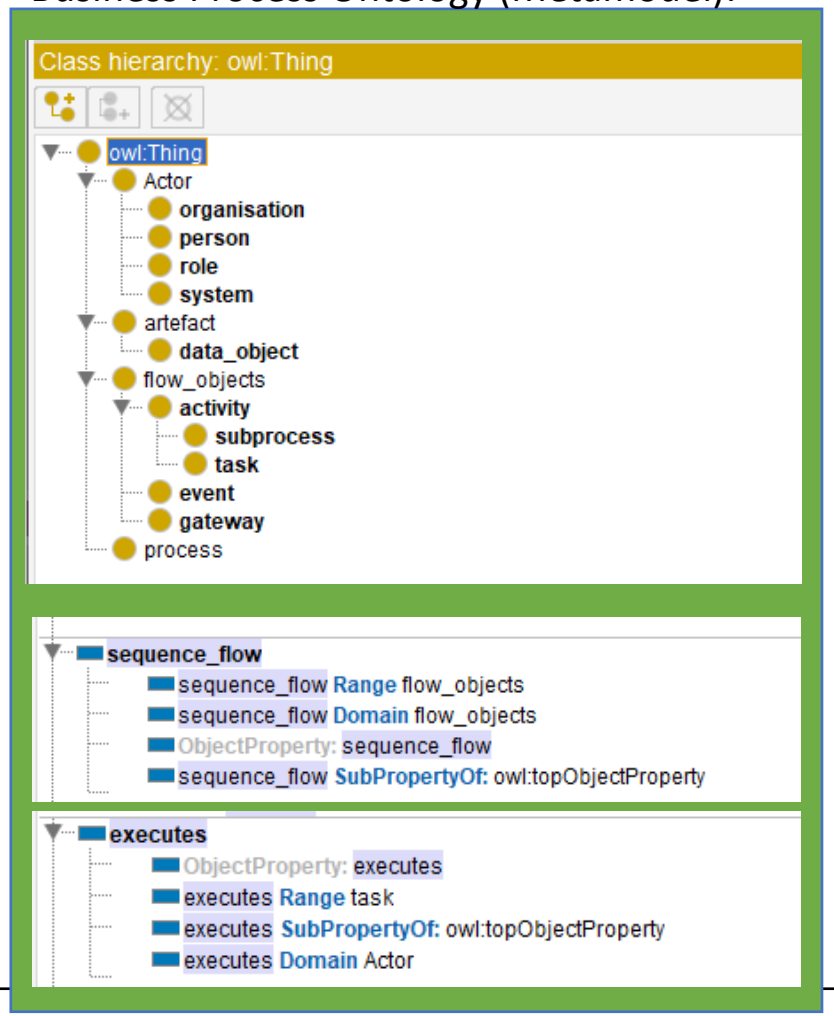



```
:Academic_Staff rdf:type owl:Class .
:lecturer rdf:type owl:Class ;
          rdfs:subClassOf :Academic_Staff .
:module rdf:type owl:Class .
:course rdf:type owl:Class ;
        rdfs:subClassOf :module .
:is_taught_by rdfs:domain :module;
              rdfs:range :lecturer .
:KE rdf:type :course ;
    :is_taught_by :knut ;
    :credits 6 ;
    :title "Knowledge Engineering" .
:knut rdf:type :lecturer ;
      :name "Knut Hinkelmann" .
```

Source: Prof. Dr. Knut Hinkelmann

Example: Concepts and Instances for Process Modelling

Business Process Ontology (Metamodel):



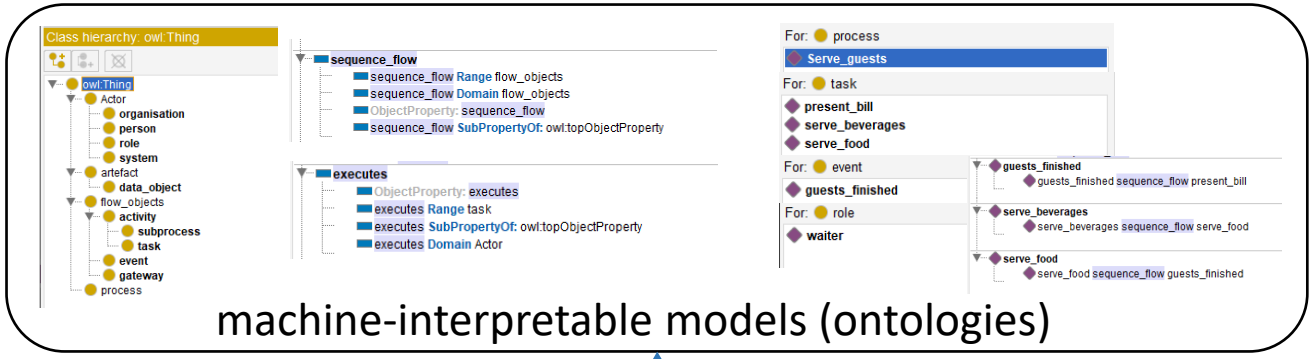
Process Model for Serve Guests

Source: Prof. Dr. Knut Hinkelmann

Knowledge Base

Reasoning/ Decision Making

Models



machine-interpretable models (ontologies)

Reality

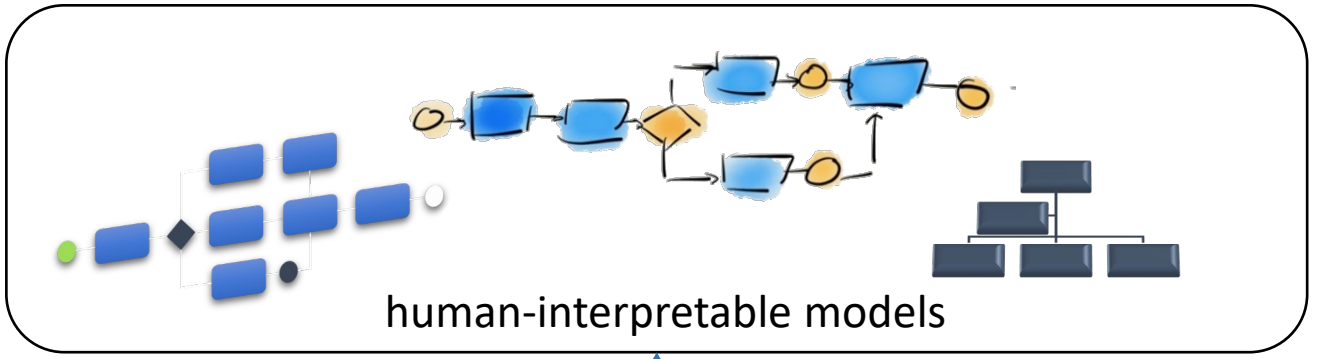
Problems of knowledge bases:

- Creating knowledge graphs is difficult for non-ontology experts or for domain experts; it requires skills in ontology languages.
- Maintaining knowledge graphs is a complex and knowledge-intensive task. Especially:
- in large organizations,
- in applications where various stakeholders with different expertise are required.

Source: Adapted from Prof. Dr. Knut Hinkelmann

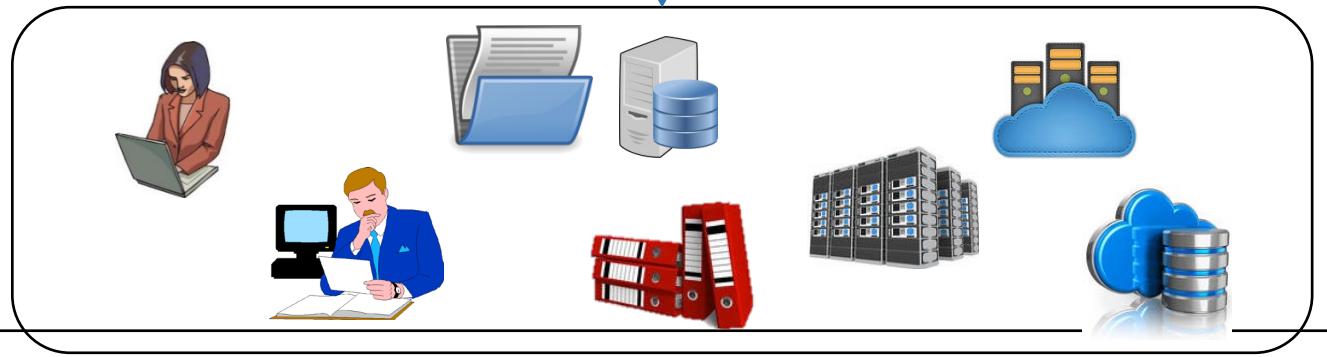
Graphical models are appropriate for humans

Communication/
Analysis/
Decision Making



Models

Reality

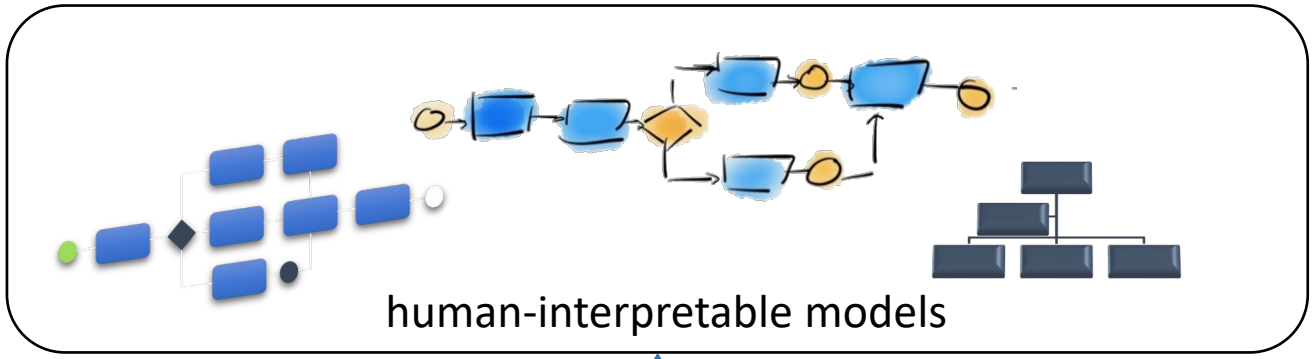


- Graphical models are conceptual models with graphical notations.

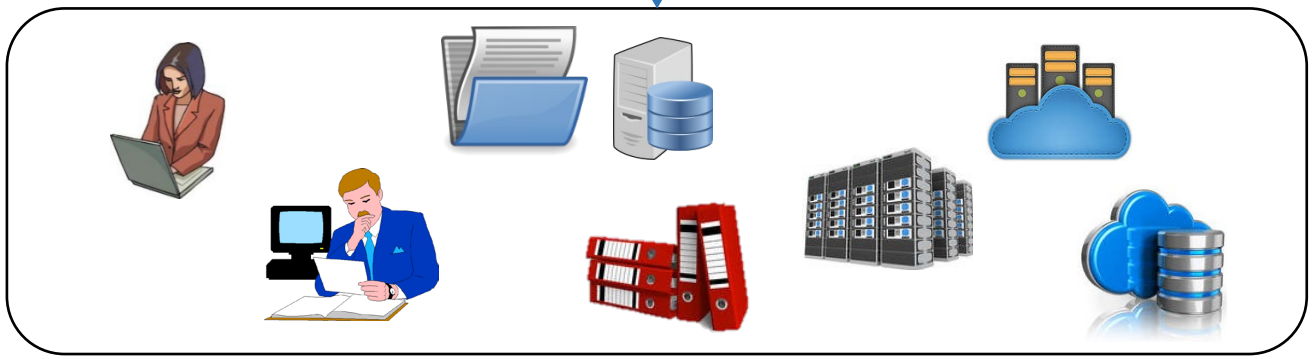
Source: Prof. Dr. Knut Hinkelmann

Humans use knowledge to interpret graphical models

Communication/ Analysis/Decision Making



Models



Reality

Source: Prof. Dr. Knut Hinkelmann

Enterprise Models

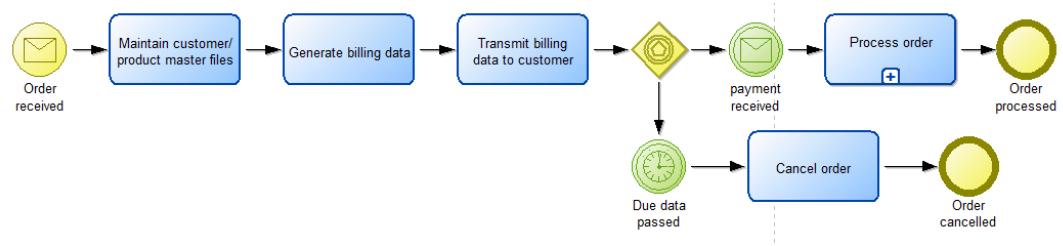
Enterprise models are graphical models that capture relevant knowledge of an enterprise…

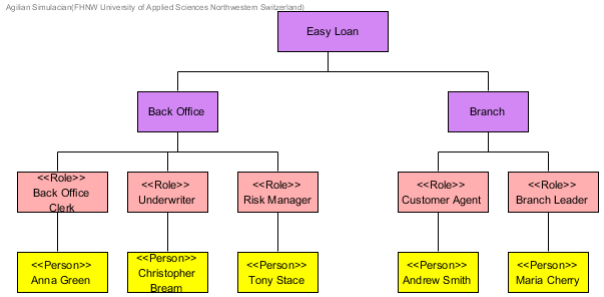
…and visualize it through a graphical notation…

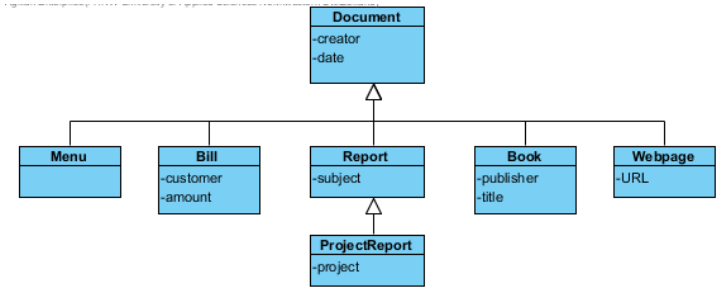
…facilitating understanding, communication, problem-solving, and meaning negotiation among humans.

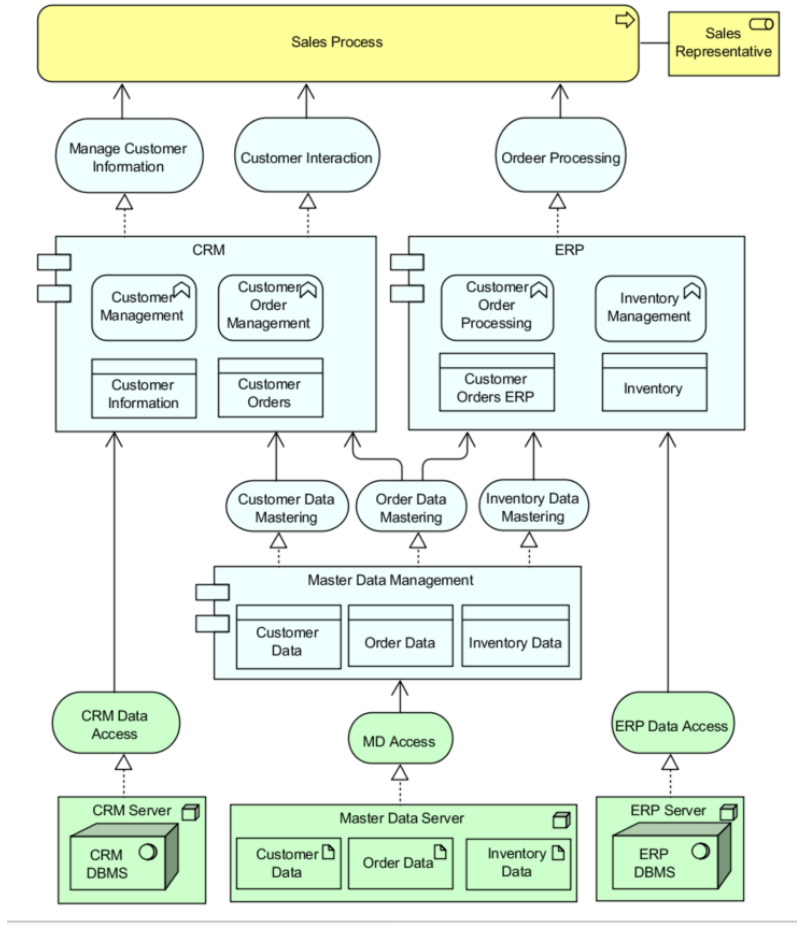
They can be used to analyze the impact of changes, cost, risk, security, compliance and other relevant KPIs.

Examples of Enterprise Models





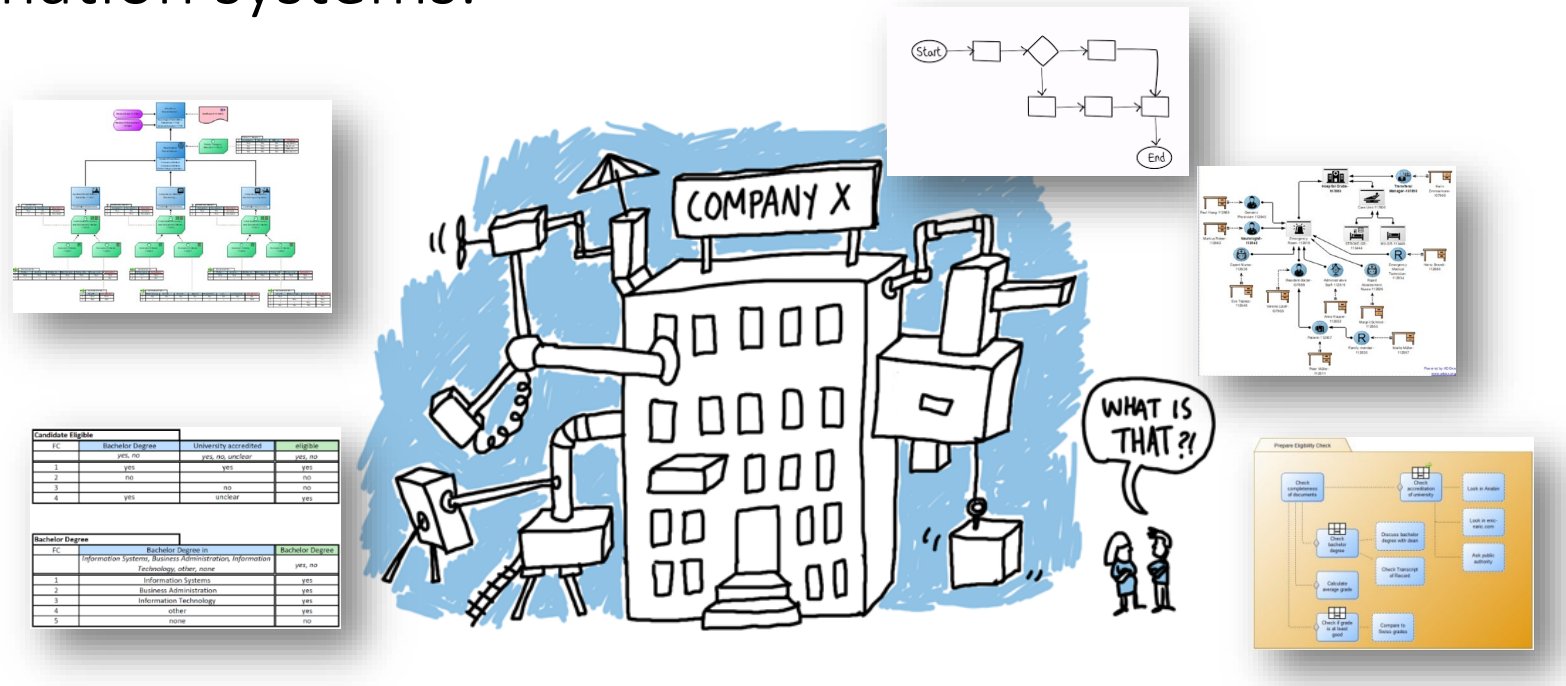




Source: Prof. Dr. Knut Hinkelmann

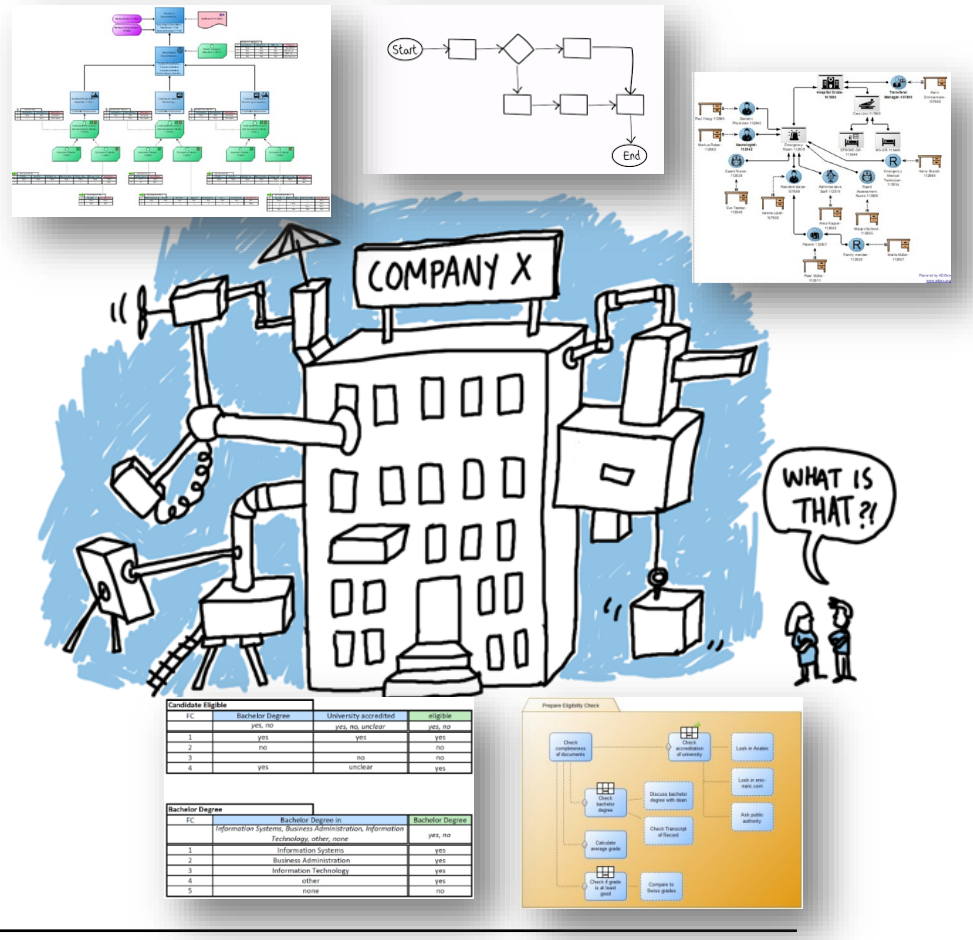
Enterprise Modelling (EM) /1

– Enterprise Modelling is an established discipline for the conceptual representation, design, implementation, and analysis of information systems.



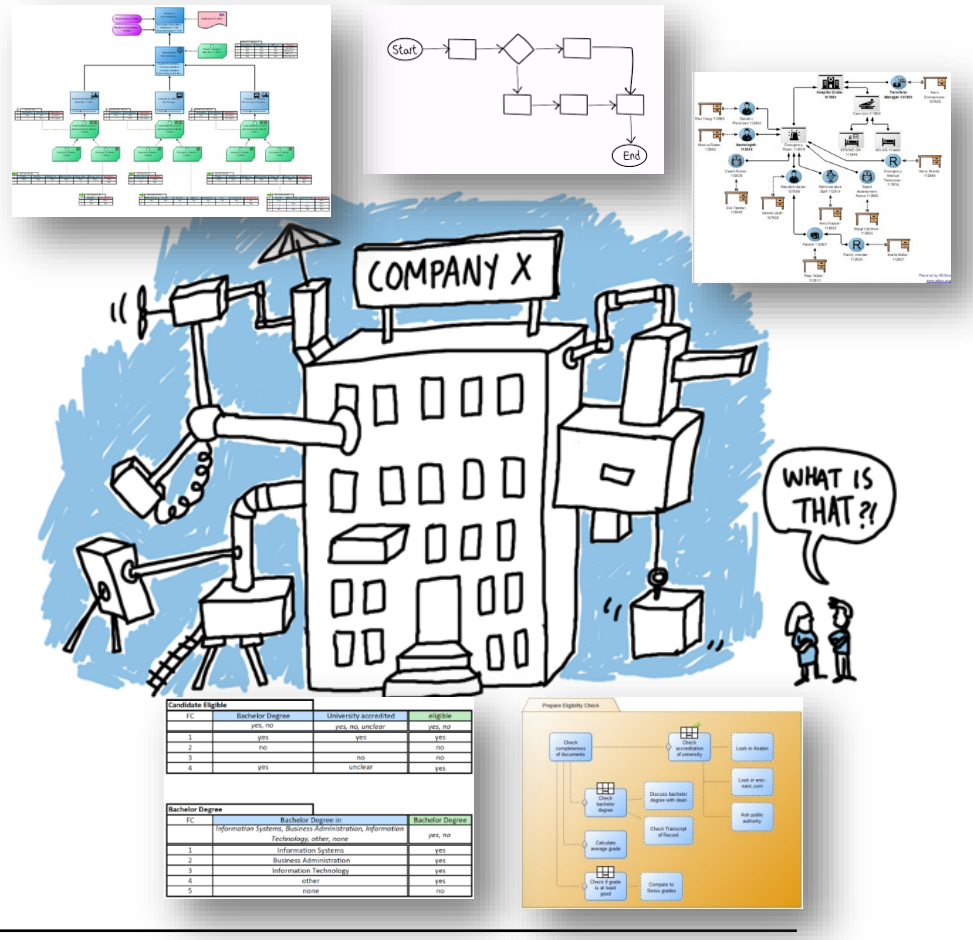
Enterprise Modelling (EM) /2

- EM offers the creation of domain-specific modeling languages (DSML) (Van Deursen et al. 2000, Frank 2014) to target specific stakeholder groups
  - Tailored graphical notations increase the shared understanding of a given domain of discourse.
- EM enables participative modelling and involve different stakeholder groups (e.g., Stirna et al. 2007)
- EM supports business and information systems engineering.



Enterprise Modelling (EM) /3

– In a practical sense, enterprise modelling refers to the act of creating or adapting enterprise models **using modelling languages.**

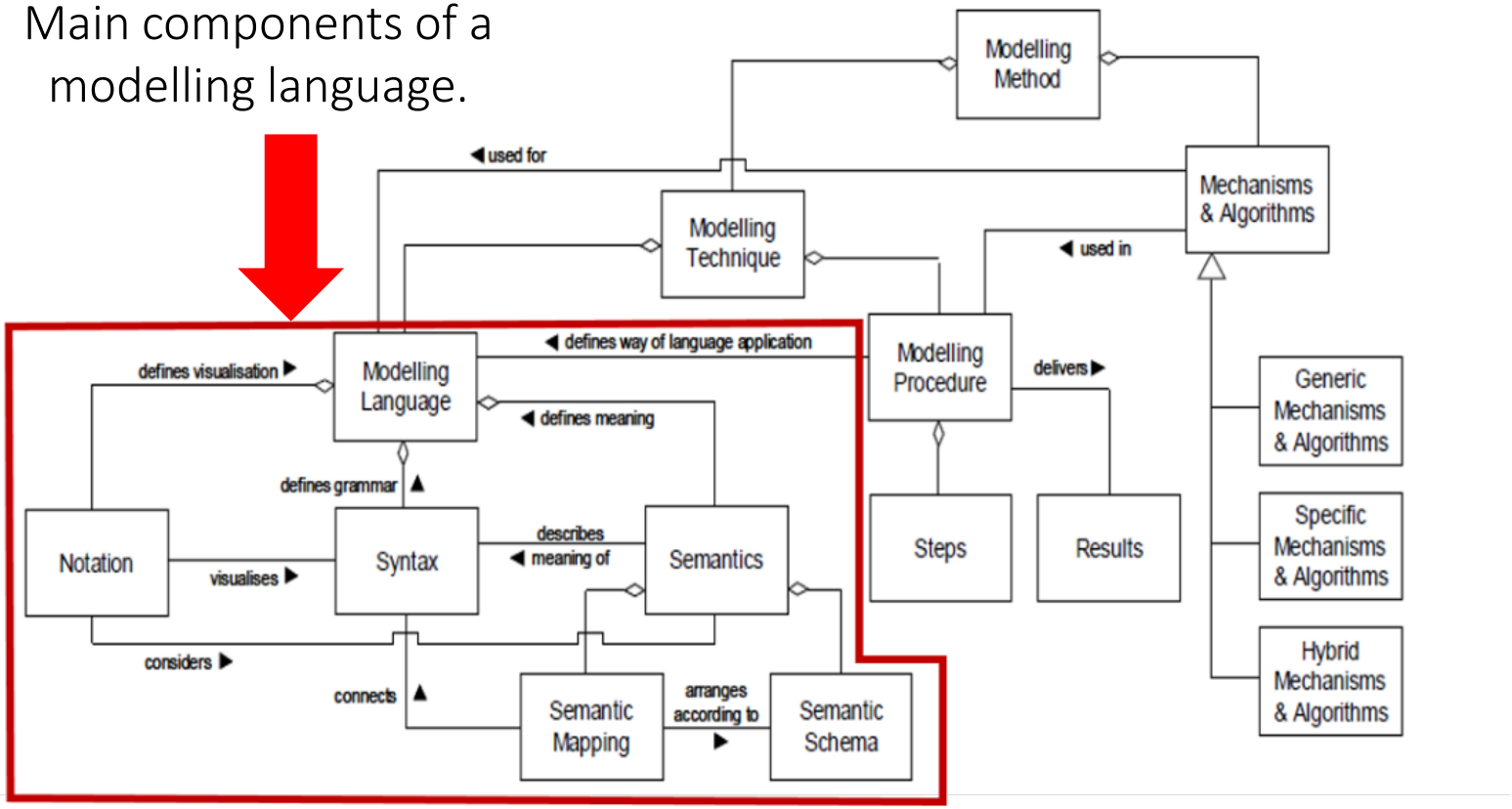


Source: Prof. Dr. Knut Hinkelmann

Modelling Language

- A modelling language consists of:
  - Graphical notation
  - Abstract Syntax
  - Semantics

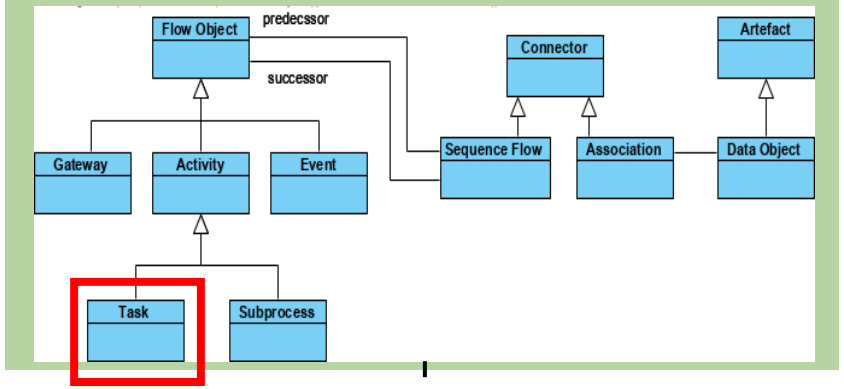
Main components of a modelling language.



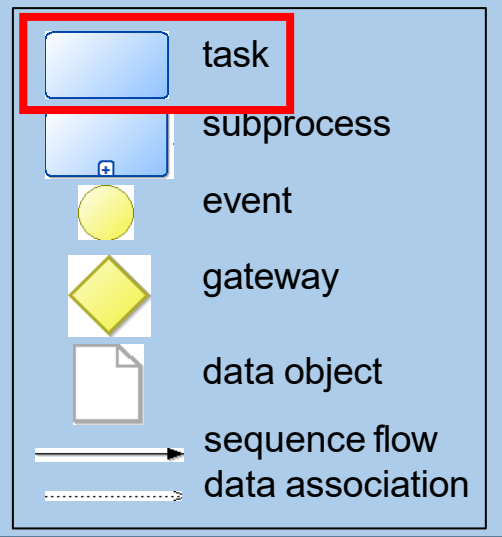
Karagiannis and Kühn (2002)

A Business Process Modelling Language Example

Abstract Syntax (Meta-model):
Concepts and relations that can be used to create models.

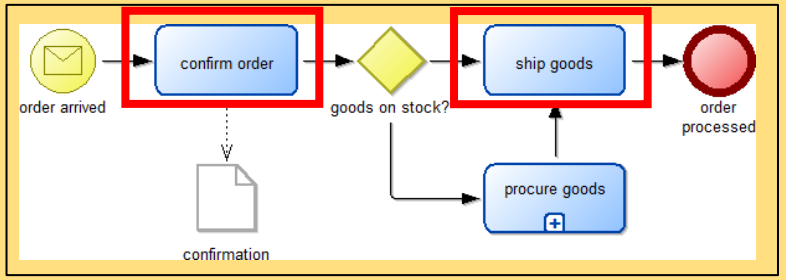


Graphical Notation:
Notation/appearance of meta-model concept.

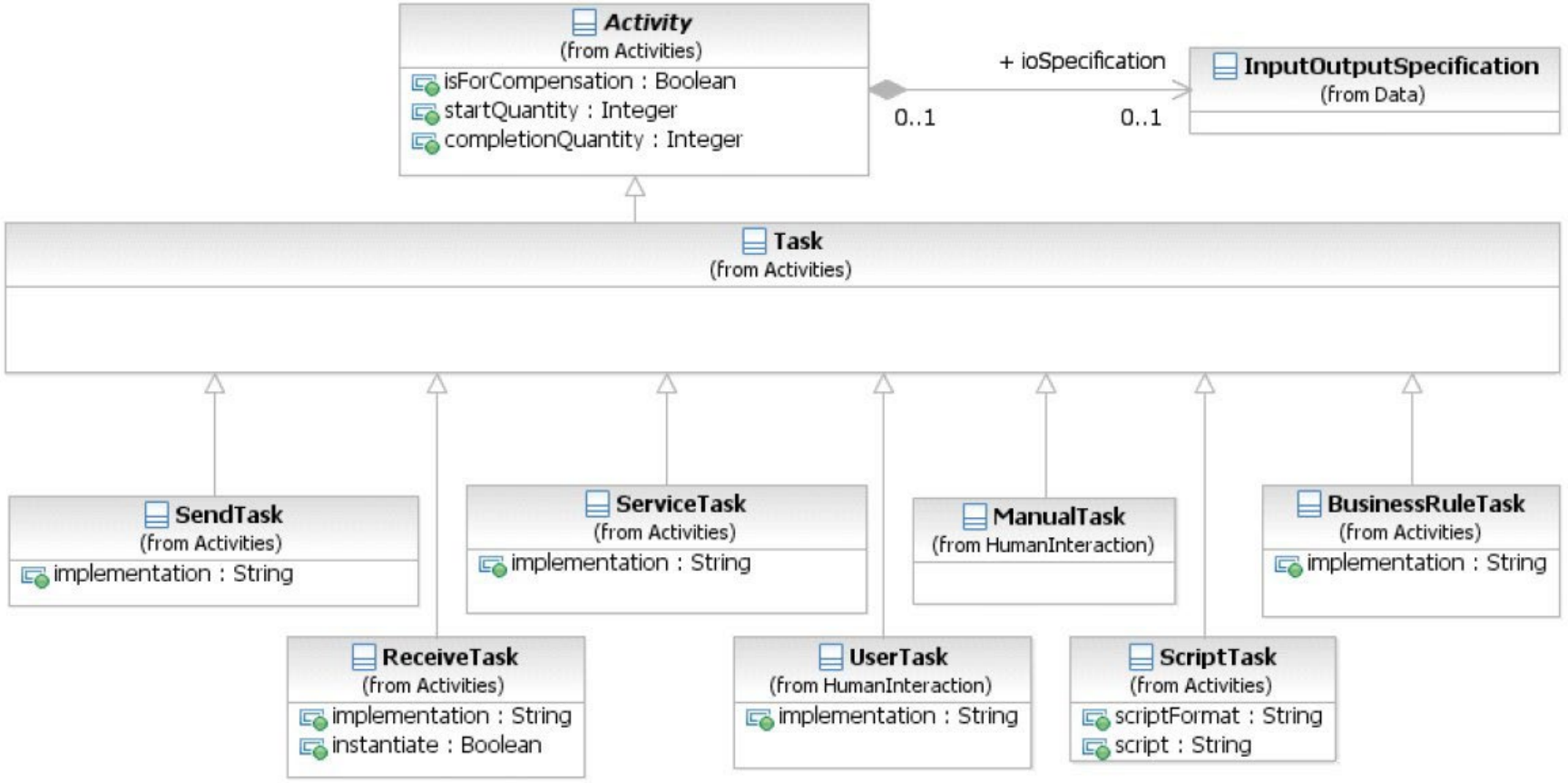


An enterprise model contains instances of the concepts defined in the meta-model.
E.g., the object „confirm order" represents a real entity; it is an instance of the concept "task"

Enterprise Model:



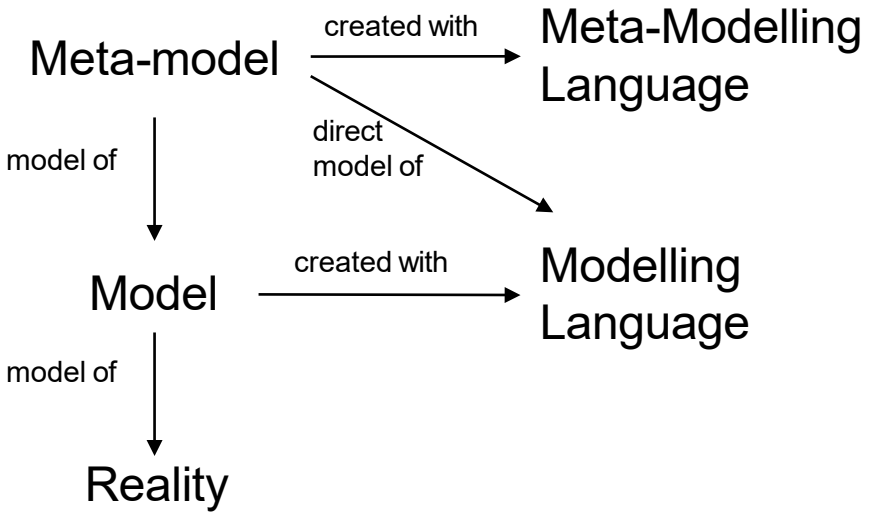
Subset of the BPMN Meta-model as a UML Class Diagram



(UML Class diagrams where originally designed for modelling in object-oriented programming. This is why they contain operations and other features, which are not relevant for most modelling languages).

Source: Prof. Dr. Knut Hinkelmann

Meta-model of a Modelling Language



Metamodelling Hierarchy (Strahringer 1996)

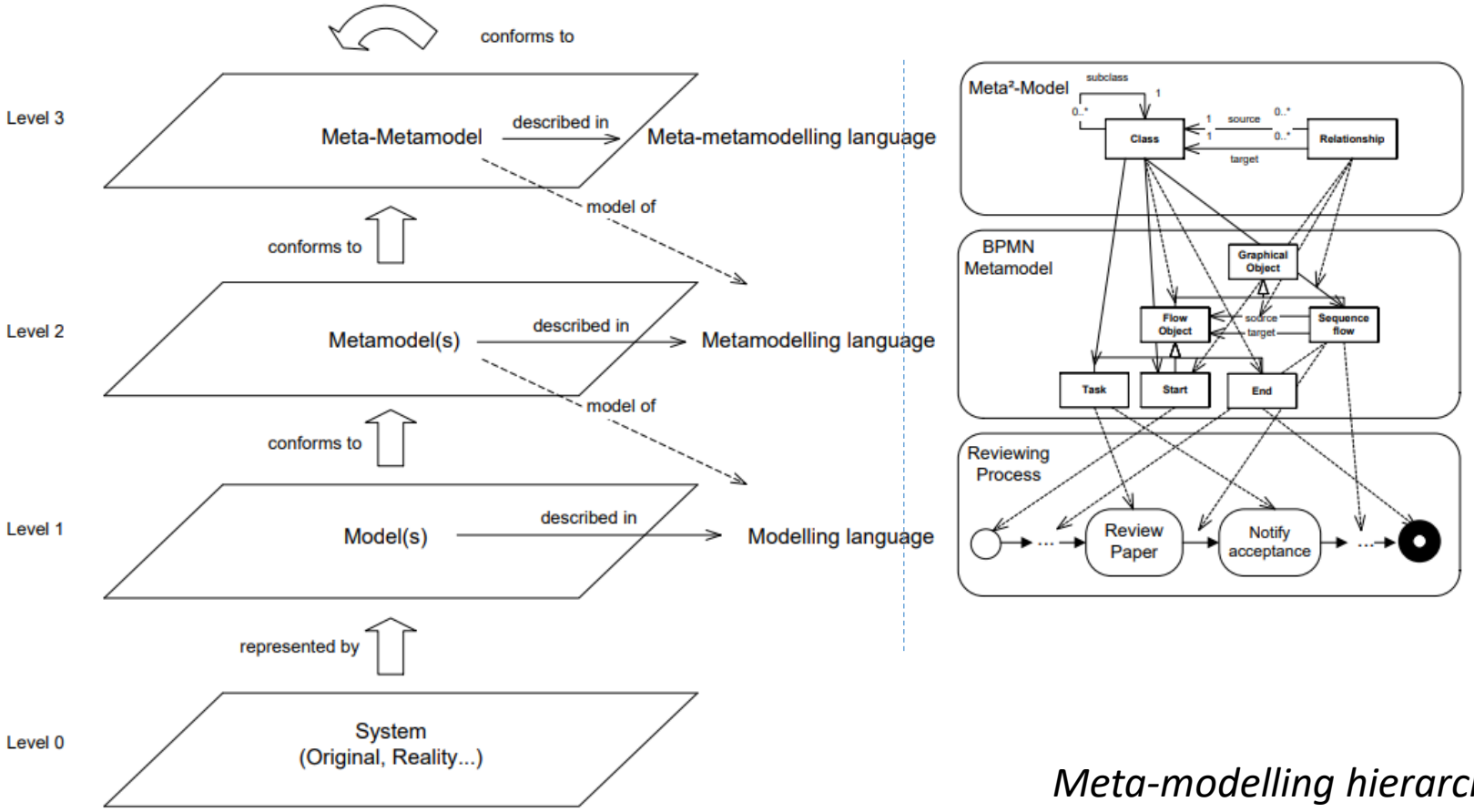
A ***meta-model*** defines the semantics of the modelling language, i.e. the building blocks that can be used to make a model. It defines the

- object types that can be used to represent a model
- relations between object types
- attributes of the object types
- rules to combine object types and relations

- The meta-model can be described in a modeling language, too. This is called the meta-modelling language.
  - UML Class Diagram is the meta-modelling language for BPMN 2.0.

Source: adapted from Prof. Dr. Knut Hinkelmann

Example: The meta-modelling hierarchy for BPMN 2.0



Meta-modelling hierarchy.
Adapted from (Efendioglu et al. 2017)

Recap: Models, Modelling, Modeling Language, Meta-Model

Enterprise Model

A reproduction of the part of the reality of an enterprise containing essential aspects to be investigated.

Enterprise Modelling

Creating enterprise models using predefined concepts.

Modelling Language

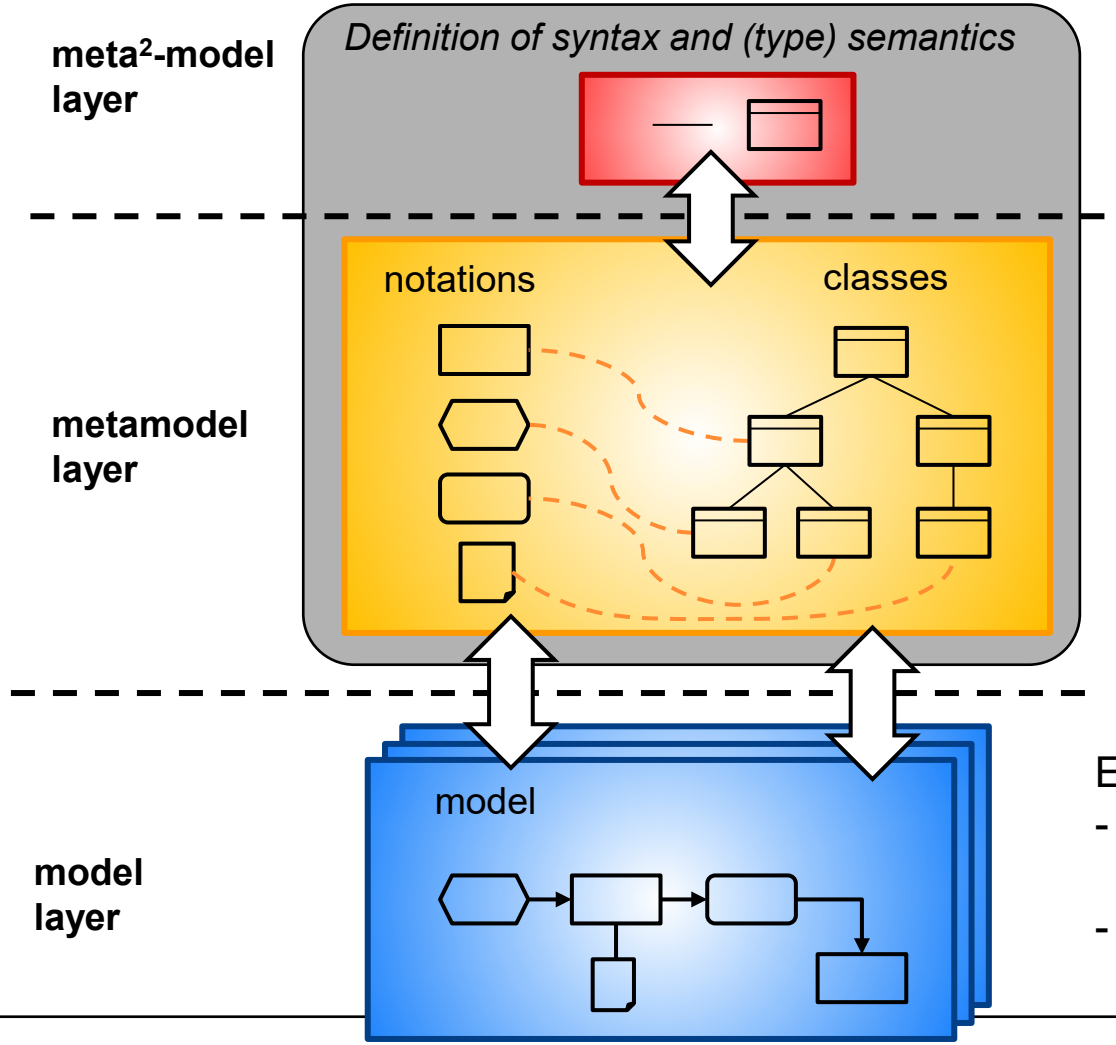
Notation/Visualization of the concepts that can be used for modeling

Meta Model

The concepts of the modeling language are predefined in a so-called meta-model

Source: Prof. Dr. Knut Hinkelmann

High-level Architecture of Meta-Modelling Tools



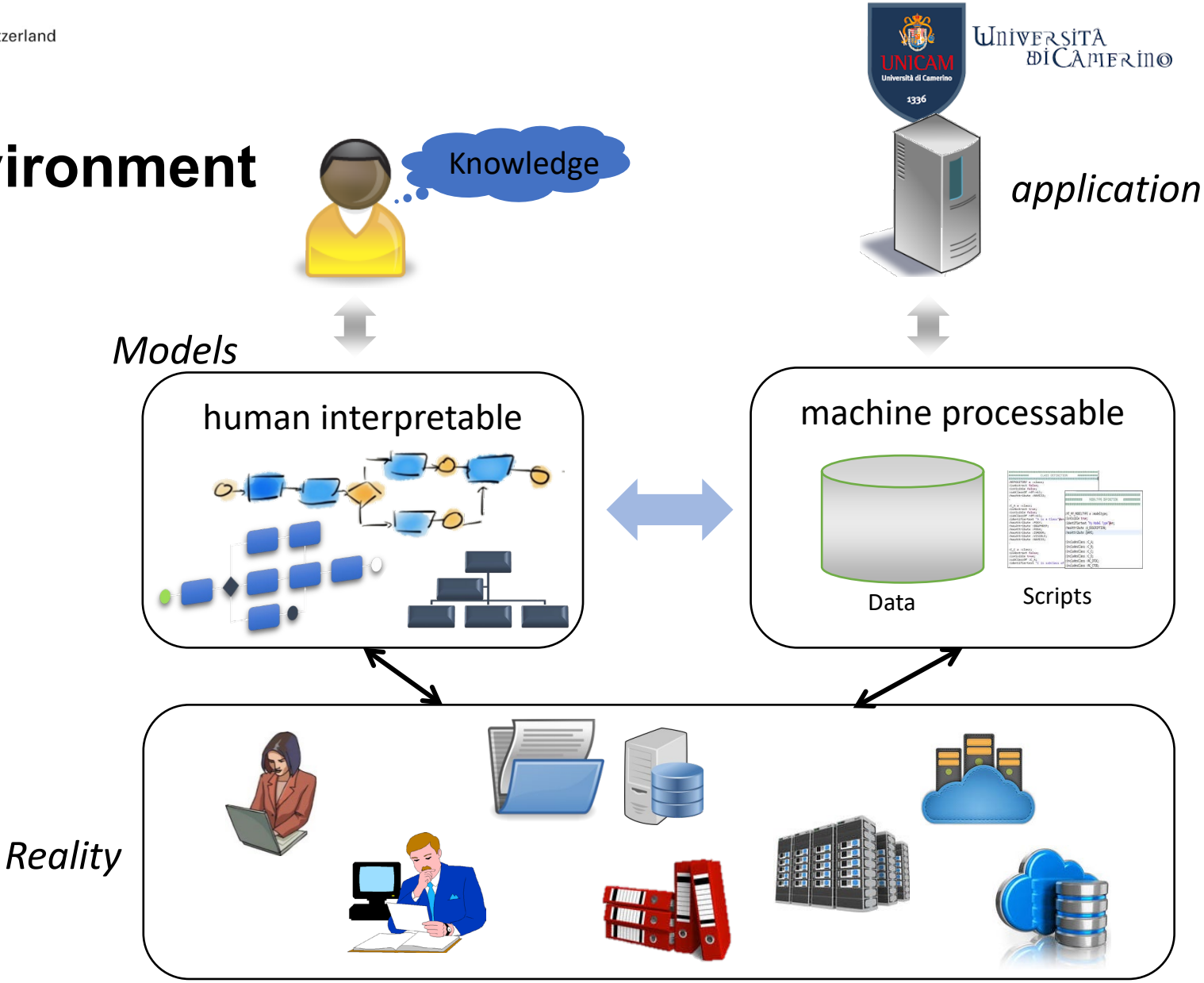
E.g.,
- ADOxx Development Toolkit
  - https://adoxx.org/
- MetaEdit+ Workbench
  - https://www.metacase.com/products.html

E.g.,
- ADOxx Modeling Toolkit
  - https://adoxx.org/
- MetaEdit+ Modeler
  - https://www.metacase.com/products.html

Meta-modelling Environment

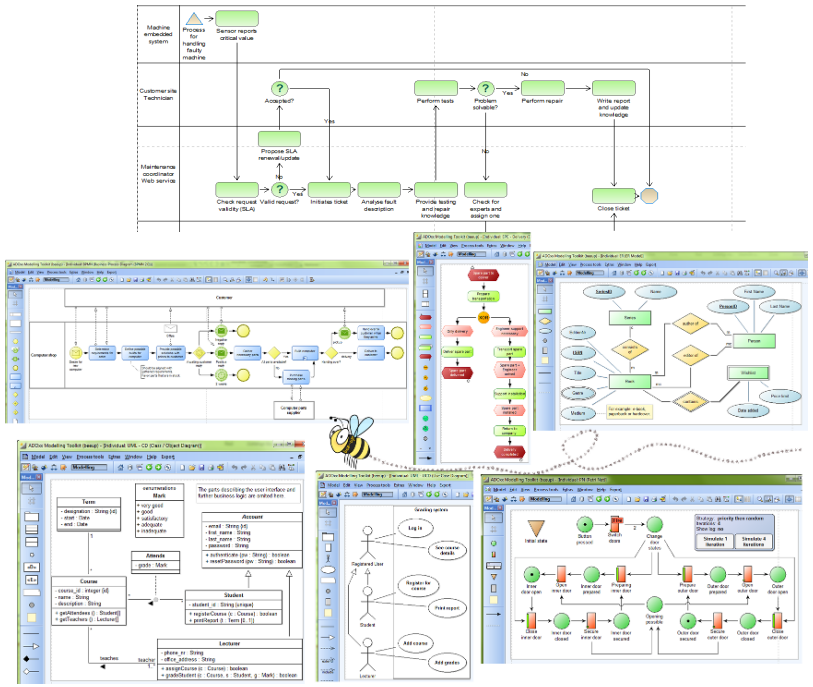
– Meta-modelling tools store graphical models and meta-model structure in databases.

Problems of metamodeling environments:
Automation limitation!
Models are not interpretable by machines like if we had ontologies.
-> No Machine Reasoning!



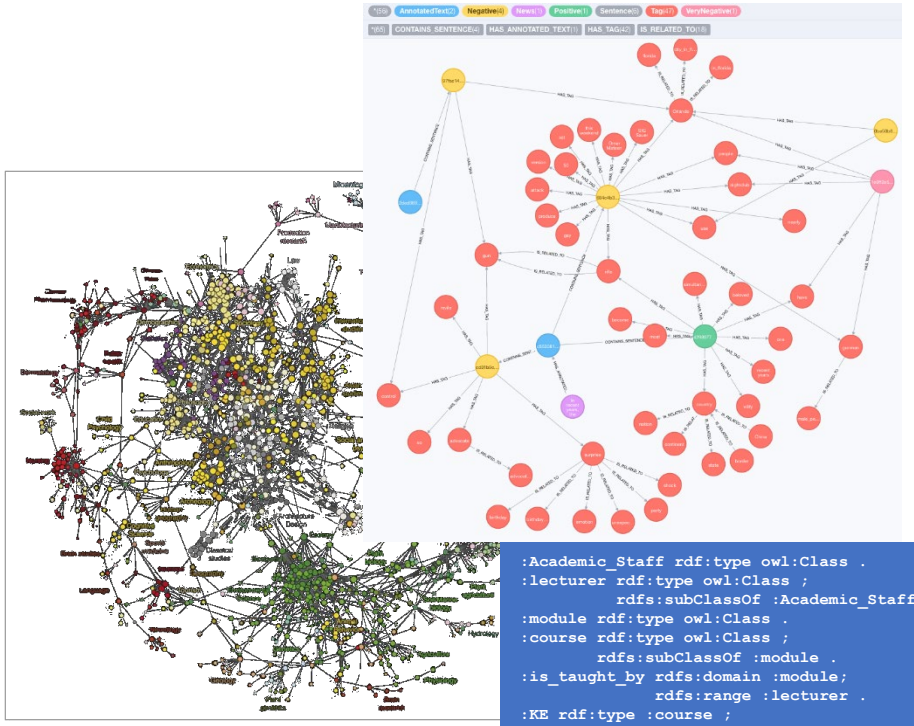
Solution: Convergence of Two Knowledge Representations

Enterprise Models



Modeling using predefined *concepts.*

Knowledge Graphs/Ontologies



```
:Academic_Staff rdf:type owl:Class .
:lecturer rdf:type owl:Class ;
          rdfs:subClassOf :Academic_Staff .
:module rdf:type owl:Class .
:course rdf:type owl:Class ;
        rdfs:subClassOf :module .
:is_taught_by rdfs:domain :module;
              rdfs:range :lecturer .
:KE rdf:type :course ;
    :is_taught_by :knut ;
    :credits 6 ;
    :title "Knowledge Engineering" .
:knut rdf:type :lecturer ;
      :name "Knut Hinkelmann" .
```


Source: Prof. Dr. Knut Hinkelmann

Convergence of two knowledge representations

- The convergence of the Enterprise modelling and Knowledge Graph/Ontologies is a recent research interest that strives to address the challenge and limitations of adopting only enterprise models and only knowledge graphs/ontologies.
- Two prominent approaches:
  - Semantic Lifting
  - Ontology-based Meta-modelling

Useful references

- Laurenzi, E. (2020): An Agile and Ontology-Aided Approach for Domain-Specific Adaptations of Modelling Languages. PhD thesis, University of Pretoria. Available at: https://repository.up.ac.za/handle/2263/73419
- Laurenzi, E., Hinkelmann, K., & van der Merwe, A. (2018). An Agile and Ontology-Aided Modelling Environment. In R. Buchmann, D. Karagiannis, & M. Kirikova (Eds.), The Practice of Enterprise Modelling. PoEM 2018. (pp. 221–237). Vienna: Springer, Cham. https://doi.org/10.1007/978-3-030-02302-7_14
- Griesinger, F.; Seybold, D.; Wesner, S.; Domaschka, J.; Woitsch, R.; Kritikos, K.; Hinkelmann, K.; Laurenzi, E.; Iranzo, J.; González, R. and Tuguran, C. (2017). BPaaS in Multi-cloud Environments - The CloudSocket Approach. In European Space Projects: Developments, Implementations and Impacts in a Changing World - EPS Porto 2017, ISBN 978-989-758-311-7, pages 50-74. DOI: 10.5220/0007901700500074
- Emmenegger, S. et al. (2017). An Ontology-Based and Case-Based Reasoning Supported Workplace Learning Approach. In Communications in Computer and Information Science. Springer, Cham, pp. 333–354. Available at: http://link.springer.com/10.1007/978-3-319-66302-9_17.
- Karagiannis, P. Burzynski, E.-T. Miron, The "IMKER" Case Study - Practice with the Bee-Up tool (2017)
- Hinkelmann, K., Laurenzi, E., et al. (2016). A Semantically-Enhanced Modelling Environment for Business Process as a Service. In 2016 4th International Conference on Enterprise Systems (ES). IEEE, pp. 143–152. Available at: http://ieeexplore.ieee.org/document/7880484/.
- Karagiannis, H.C. Mayr, J. Mylopoulos, Domain-Specific Conceptual Modeling, Springer (2016)
- Karagiannis, R. Buchmann, P. Burzynski, U. Reimer, M. Walch, Fundamental Conceptual Modeling Languages in OMiLAB, Springer (2016)
- Hinkelmann, K., Gerber, A., Karagiannis, D., Thönssen, B., Van der Merwe, A., Woitsch, R. (2016). A new paradigm for the continuous alignment of business and IT: Combining enterprise architecture modelling and enterprise ontology. Computers in Industry. Vol. 79, pp. 77-86. Available at: http://www.sciencedirect.com/science/article/pii/S0166361515300270.
- Karagiannis (2015): Agile Modeling Method Engineering, 19th Panhellenic Conference on Informatics (PCI) 2015, 01.-03.10.2015, Athens, Greece, ACM NY USA, DOI: http://dx.doi.org/10.1145/2801948.2802040
- Azzini, A., Braghin, C., Damiani, E., & Zavatarelli, F. (2013). Using Semantic Lifting for improving Process Mining: A Data Loss Prevention System Case Study. In SIMPDA 2013: 3rd International Symposium on Data-driven Process Discovery and Analysis, 62-73. CEURWS.org
- Process Mining: A Data Loss Prevention System Case Study. In SIMPDA 2013: 3rd International Symposium on Data-driven Process Discovery and Analysis, 62-73. CEURWS.org.
- Hrgovcic, V., Karagiannis, D., & Woitsch, R. (2013). Conceptual Modeling of the Organisational Aspects for Distributed Applications: The Semantic Lifting Approach. In 2013 IEEE 37th Annual Computer Software and Applications Conference Workshops (pp. 145–150). IEEE. https://doi.org/10.1109/COMPSACW.2013.17
- Atkinson, C., & Kuhne, T. (2003). Model-Driven Development: A Metamodeling Foundation. IEEE Software, 20(5), 36–41. https://doi.org/10.1109/MS.2003.1231149
- Karagiannis and H. Kühn, “Metamodelling Platforms,” Proceedings of EC-Web 2002 – DEXA 2002, LNCS,Springer, vol. 2455, pp. 182–182, 2002.