Philosophical Foundations of Conceptual Models

John Mylopoulos
University of Ottawa

Centro de Informática, FUPE
Recife, November 20, 2019

Acknowledgements

- This is joint work with Nicola Guarino and Giancarlo Guizzardi. My views on Conceptual Modelling since the early 1970s were heavily influenced by ideas of Alex Borgida (RutgersU, emeritus) and Hector Levesque (UToronto, emeritus).

Abstract

We review and analyze alternative answers to fundamental questions for Conceptual Modelling, such as: (a) What is a conceptual model? (b) Among models used in Computer Science, which are conceptual and which are not?(c) How are conceptual models different from other models used in the Sciences and Engineering? Our study draws from literature in Philosophy, Cognitive Science, Logics, as well as several areas within Computer Science, including Databases, Software Engineering (SE), Artificial Intelligence (AI), Information Systems Engineering (ISE), among others.

Motivation

- A recent ER paper [Delcambre18] proposes “A Reference Framework for Conceptual Modelling” that attempts to answer fundamental questions about Conceptual Modelling (CM), such as: What is and isn’t a conceptual model (cm)? Who does conceptual modelling? Etc.
- We applaud the efforts of our colleagues, this discussion is long overdue!
- For CM to be a respectable research area, it needs to reach consensus on what cm and conceptual modelling are, and what are the research questions that need to be addressed.
- This presentation draws on [Guarino19].

Conceptual models

- It is a basic tenet of Cognitive Science and Philosophy of Mind that cognitive processes create, use and transform ***mental representations*** of the world [SEP05].
- These representations can be conceptual (consisting of concepts), for example thoughts, or non-conceptual, e.g., sensations (aka tacit knowledge). We call the former ***conceptual mental representations***.
- ***Conceptual models*** are ***computational*** models of conceptual mental representations, to be used for purposes of understanding, communication and problem solving.
- Conceptual models are different from other kinds of models used in Science and Engineering in that they don't model the world, but rather ***our conceptualizations of the world***.

Why computational?

- Conceptual models are computational in that they are stored in computers and are analyzed or reasoned with through computational means.
- Computationality renders conceptual models scalable and analyzable. Pragmatically speaking, it is inconceivable that conceptual models would be useful “for purposes of understanding, communication and problem solving” if they were not computational.
- It is no accident that they came about with the advent of computers.

Concepts as atoms of thought

- Concepts are the atoms of conceptual mental representations. They are formed through experience (Locke, Hume), including social interaction, and provide a “lens for looking at the world”, or “a language of thought” [Fodor75].
- There seems to be consensus that concepts come with a definitional structure, they are ***associated*** to other concepts (associationism, see [SEP18]) and have instances/referents (see Frege et al).

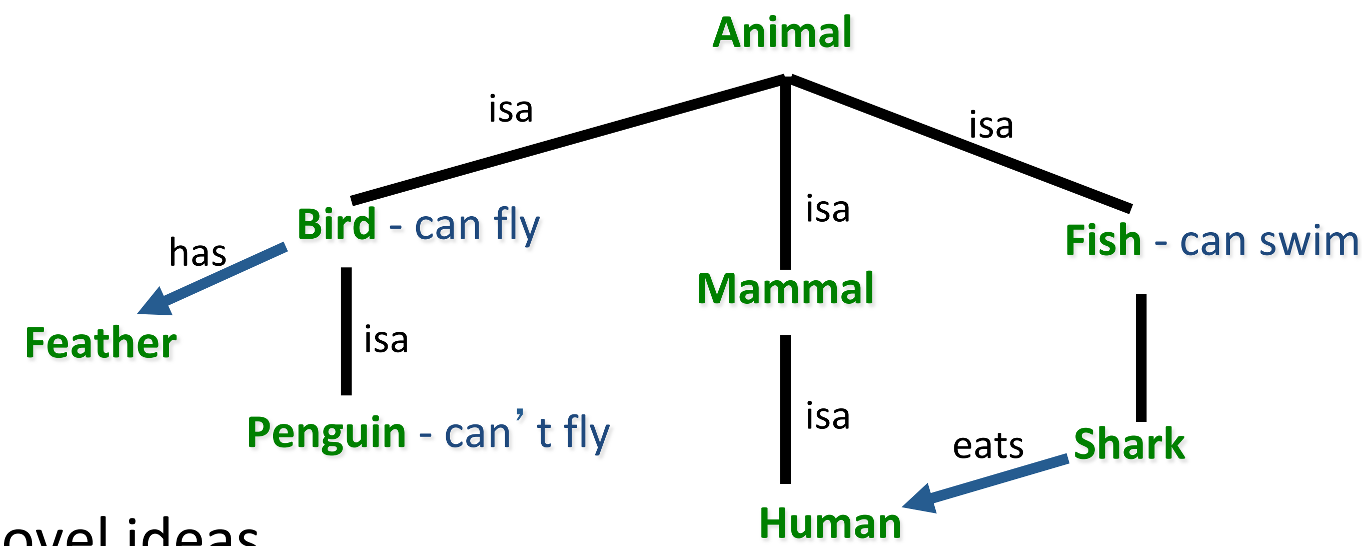
Concepts and associations

- Concepts have been a topic of study in Philosophy since Plato and Aristotle; Associationism has been studied at least since the empiricists (Locke, Hume).
- Together they constitute the philosophical foundations of Conceptual Modelling.
- Conceptual modelling languages adopt this perspective: semantic networks, semantic data models, description logics, OO models, business process models, …
- Others modelling frameworks adopt a logical perspective: Concepts have a propositional, rather than an associationist, structure.

Origins of CM in Computer Science

- In AI, Ross Quillian proposed *semantic networks* as a model of the structure of human memory [Quillian66].
- In Programming Languages, Ole-Johan Dahl proposed *Simula*, an extension of ALGOL 60, for simulation programs that require "world modeling" [Dahl68].
- In Databases, Jean-Raymond Abrial proposed a *semantic model* in 1974 [Abrial74], shortly followed by Peter Chen's *entity-relationship model* [Chen76] as advances over logical data models, such as Codd's relational model.
- Doug Ross proposed in the mid-70s the *Structured Analysis and Design Technique* (*SADT*) as a "language for communicating ideas" [Ross77]. The technique was used to specify requirements for software systems.

Semantic networks (1966)



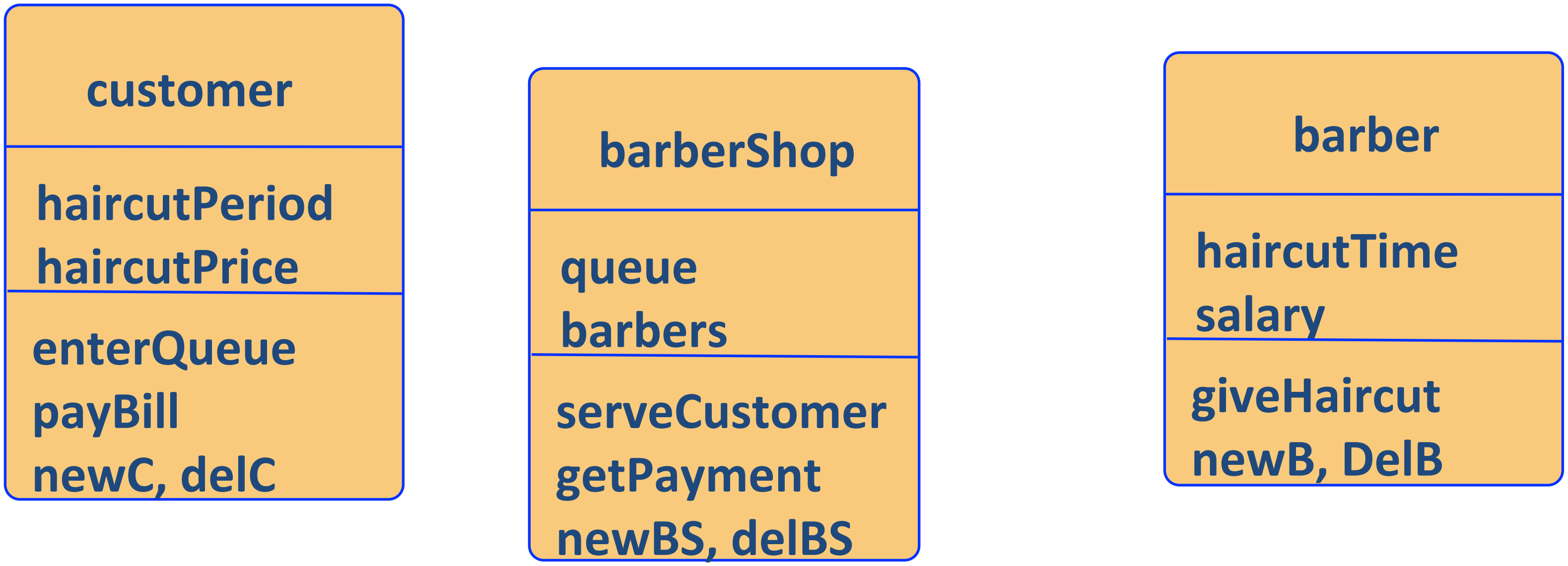
Novel ideas

- Models are built out of *concepts* and *associations*
- *Inheritance of attributes* -- default, single
- Computation defined in terms of *spreading activation* -- e.g., discovering the meaning of "horse food"

    horse --> animal --> eat --> food

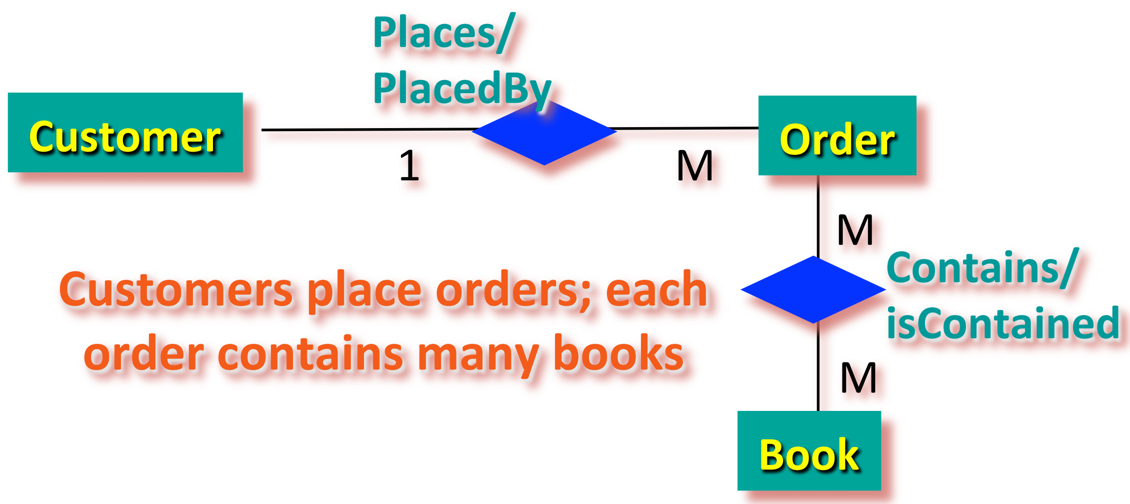
    horse --> animal --> madeOf --> meat --> food

Simula (1967)



- Ole-Johan Dahl proposed it as an extension of ALGOL 60, for simulation applications.
- A (simulation) program consists of classes and instances.
- Instances ***model the simulated domain***, classes define common features of instances, are organized into subclass hierarchies.

Entity-Relationship Model (1975)



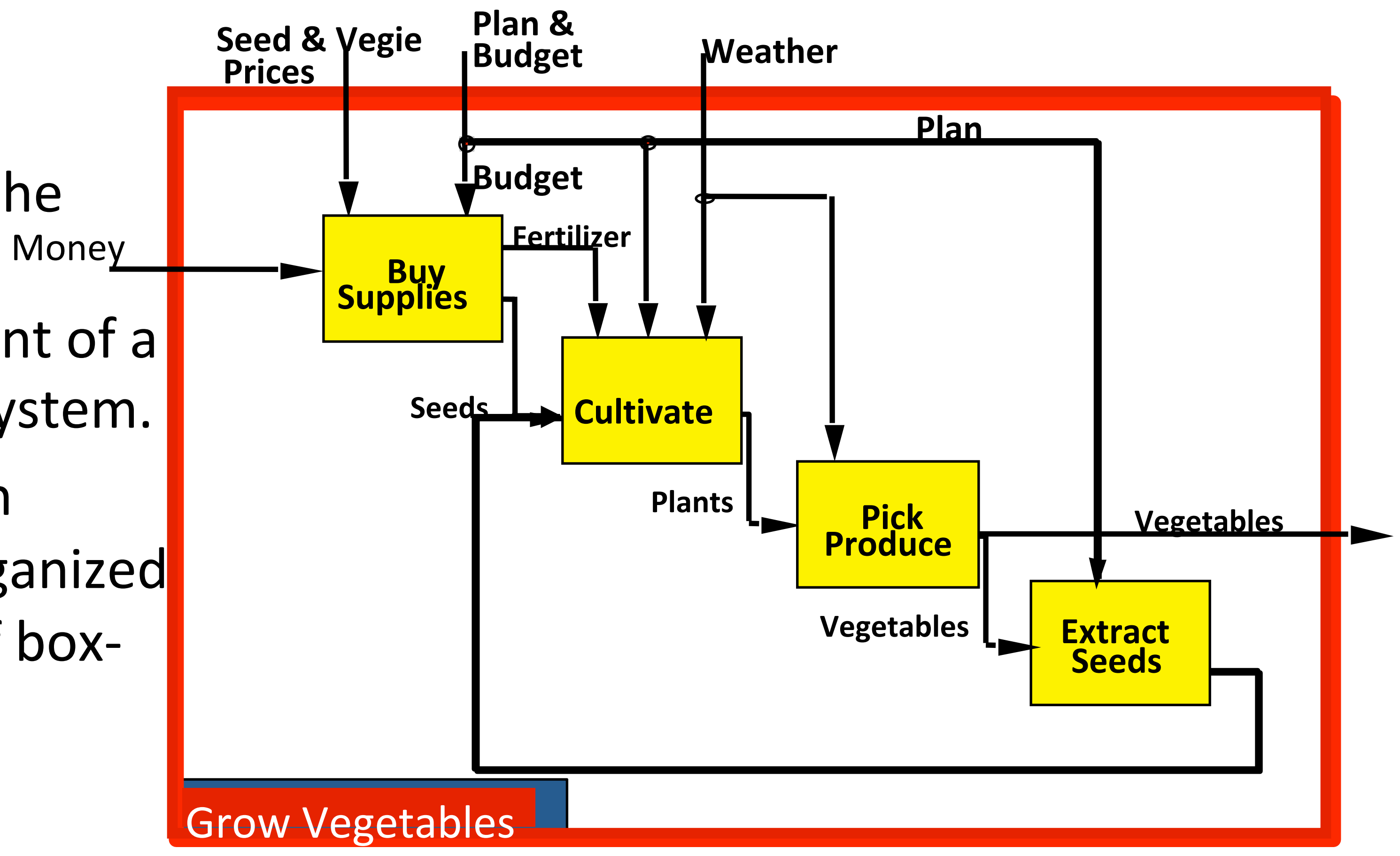
Novel ideas

- Assumes that application consists of ***entities*** and ***relationships*** (*ontological assumptions*)
- Shows how a conceptual schema can be mapped onto a logical one.
- [Abrial's semantic model was more akin to OO data models, but did offer entities and relations too]

Structured Analysis and Design Technique (SADT)

Novel Ideas

- ✓ Modeling the operating environment of a software system.
- ✓ Application models organized in terms of box-inside-box notation.



History of Conceptual Modelling

- In AI, semantic networks served as foundation for knowledge representation languages, e.g., KLONE [Brachman78] and [Levesque79], leading to Description Logics [Baader03] and OWL, a WWW standard.
- Dahl's Simula included ***classes*** and their ***instances*** for modelling a domain, evolved into Smalltalk at Xerox PARC, and led to OO Modelling and UML.
- In Databases, the Extended ER Model and UML are used for conceptual schema design.
- In Requirements Engineering, SADT was followed by formal modelling languages, e.g., [Greenspan82] and Goal-Oriented Languages, e.g., KAOS [Dardenne93], i* [Yu97].
- New research threads on Conceptual Modelling have sprang in Business Process Management (BPM), Enterprise Architectures (EAs) and more.

Common thread

- The common thread that underlies all these efforts is an attempt to build models that view a domain "the way we do" and are consequently "direct and natural" models [Hammer78], "capture true semantics" [Roussopoulos75].
- In AI and Knowledge Representation (KR) these models serve as ***knowledge bases*** for artificial agents to conduct problem solving.
- In Databases, SE, BPM, EA they serve as useful design models for databases, software, business processes, enterprises, etc.
- Important difference between KR and other areas of Conceptual Modelling: KR places great emphasis on ***reasoning*** with a conceptual model, whereas other areas focus on using the right ontology to build the right model.

But how do we know anything about our conceptualizations?

- From ***things we say*** about our conceptual mental representations.
- For example, if you ask a database administrator to describe for you the contents of her database and she tells you that it's about students taking courses, having marks, etc., you can come up with an EER schema for it*,**.

____

* But not a relational schema, because ***that*** is not a conceptual model.

** Actually, Databases textbooks do teach how to go from a natural language description of the contents of a database to a conceptual schema, e.g., [Atzeni99].

Models in Science and Engineering

- They are usually models of the spatio-temporal properties of physical things, at least for Physical Sciences.
- They are ***objective*** in that they model the domain, not someone's conceptualization of the domain. Moreover, they are not conceptual in that they don't include in their model the concepts they use.
- Take a model of a building: Engineers use measuring instruments to construct a 3D model, but there is no representation of the concepts 'wall', window', etc.
- In CM, we may build a cm for a building by asking 10 occupants to describe their conceptualizations of the building.

Conceptual models in *some* Sciences and Engineering

- Conceptual models offer subjective views of the world.
- Some Sciences need such models, including Cognitive Science, Sociology, Law, Economics/Management Sciences, and Computer Science, to name a few.
- Same for Engineering: Much of Engineering has to do with the design of artifacts [Simon69]. And conceptual models have proven themselves essential tools for doing design.

Newton’s second law, conceptually and otherwise

- The law states that F = M * A, where F stands for ‘force’, M for ‘mass’ and A for ‘acceleration’.
- But where are the concepts? Obviously, M and A are qualities of a ***physical object*** o, while F represents an ***action*** (push) a applied to that object, and the law is relating the strength of the action to the object’s physical properties.
- A conceptual model of this law would include entity class `PhysObject` with attributes `mass`, `velocity` and `acceleration`, action class `ApplyForce` with `object` and `strength` attributes.

Completeness of conceptual models

- Since conceptual models are to be used for understanding and communication, they must satisfy two completeness properties:
  - ✓ Every instance must be associated to relevant classes it is an instance of (***no instance without its classes***);
  - ✓ Every quality attribute should be associated to the object it is an attribute of (***no attribute without its object***).
- Conceptual models are actually used in Physics for pedagogical purposes [MacKay19], but not for defining physical laws ···

Concepts must have coherence

- Many conceptual modelling languages model concepts as unary predicates and use the connectives of Logic to define new concepts. For example, I may represent concepts Man and Unmarried with predicates $\mathbb{M}$ and $\mathbb{U}$ and then define the concept $\mathbb{B}$(achelor) as $\mathbb{M} \wedge \mathbb{U}$.
- According to this view, $\neg\mathbb{M}$ and $\mathbb{M} \vee \mathbb{N}$ are concepts too. But are they? Concepts are supposed to capture the ***common properties*** of their instances (coherence). What common properties do the instances of $\neg\mathbb{M}$, or $\mathbb{M} \vee \mathbb{N}$ have?
- Concepts must have intensional properties shared by all/ most of their instances, as opposed to not having properties.
- Accordingly, concepts such as $\mathbb{ANY}$ (with everything as an instance), and expressions such as $\neg\mathbb{C}$ and $\mathbb{C}1 \vee \mathbb{C}2$ are suspicious concepts for any conceptual modelling language.

Conceptual models and ontologies

- In general, conceptual models may only include classes/ universals (e.g., conceptual schemas); but if they include tokens/particulars they also need to include their classes.
- Ontologies are shared computational models of conceptuali-zations for a given domain (e.g., universities or banks).
- Every conceptual model adopts an ontology, but only some conceptual models are ontologies.

Conceptual models as artifacts

- Conceptual models are artifacts, and like all artifacts, must have ***requirements*** and ***stakeholders***.
- For example, say we want an enterprise model of your Centre (CdI), covering strategic objectives, academic programmes, research activities, and academic staff; also it should be ≥99% complete and 100% valid (no false information!)
- Who are the stakeholders? Members of CdI with different areas of expertise, also modellers.
- Another example: Produce a specification for a meeting scheduling system for CdI, given initial requirements R; the specification should be complete, unambiguous, consistent and minimal (no unnecessary requirements).
- Stakeholders: Members of CdI, requirements engineers, modellers.

Conceptual models as *social* artifacts

- If conceptual models are computational models of conceptual mental representations, whose mental representations are they capturing?
- An individual's? Perhaps in Art, but not in Science and Engineering.
- In Computer Science conceptual models capture ***shared mental representations*** of a group of ***stakeholders***.
- It follows that conceptual modelling should be a dialectic process involving stakeholders, and conceptual models should come about through an argument among stakeholders.

Implications for the research discipline

- Perhaps there should be a superconference on Conceptual Modelling that covers the areas of Conceptual Modelling, Knowledge Representation, Models, Requirements Modelling, Business Process Modelling, Enterprise Modelling.
- For such a conference to succeed, it needs to have a programme committee that can actually evaluate papers covering all these areas.

Implications for research

- Ideas that shaped and are still shaping Conceptual Modelling have been debated for millennia in Philosophy. Paying attention to that rich history of ideas is useful in two ways:
  - ✓ It can tell us that some of our ideas won't actually work, e.g., concept definitions;
  - ✓ It can give us alternatives we should consider, e.g., concepts as prototypes.
- Looking at Philosophy and Cognitive Science allows us to define conceptual models in terms of things that no one has ever seen (conceptual mental representations), but are nevertheless accepted as things that actually exist and play a critical role in cognition.

Conclusions

- Conceptual models were initially proposed in several areas of Computer Science as direct and natural models of a domain.
- They constitute one of the most important contributions of Computer Science to Science and other disciplines in that they offer a subjective modelling technique, with potential applications in Cognitive Science, Sociology, Law, Economics/ Management and more.
- The topic of Conceptual Modelling should be included not only at University, but also Elementary and High School curricula because it offers a new kind of Math for modelling and reasoning about the world.

References

- [Abrial72] Abrial J-R., "Data Semantics", in Klimbie J. and Koffeman K. (eds) *Data Management Systems*, North-Holland, 1972.
- [Atzeni99] Atzeni P., Ceri S., Paraboschi S., Torlone R., *Database Systems: Concepts, Languages and Architectures*, The McGraw-Hill Companies, 1999.
- [Baader03] Baader F., Calvanese D., McGuiness D., *The Description Logic Handbook: Theory, Implementation, Applications*, Cambridge University Press, 2003.
- [Brachman77] Brachman, R., "What's in a concept: Structural foundations for semantic networks". *International Journal of Man-Machine Studies 9(2*), 127–152, 1977.
- [Chen76] Chen P., "The Entity-Relationship Model: Towards a Unified View of Data", *ACM Transactions on Database Systems 1(1)*, 9-36, 1976.
- [Dahl68] Dahl O-J., Myhrhaug B., "Some Features of the Simula 67 Language" 2nd Conference on Applications of Simulations. 29–31, 1968.
- [Dardenne93] Dardenne A., van Lamsweerde A., Fickas S., "Goal–directed Requirements Acquisition", *Science of Computer Programming, 20*, 1993.
- [Delcambre18] Delcambre L., Liddle S., Pastor O., Storey V., "A Reference Framework for Conceptual Modeling", International Conference on Conceptual Modeling, Springer, 2018.
- [Fodor75] Fodor, J. (1975)*. The Language of Thought,* Cambridge, MA: Harvard University Press.
- [Greenspan82] Greenspan S., Mylopoulos J., Borgida A., "Capturing More World Knowledge in the Requirements Specification", 6th International Conference on Software Engineering, Kyoto, 1982.
- [Guarino19] Guarino N., Guizzardi G., Mylopoulos J., "On the Philosophical Foundations of Conceptual Models", submitted for publication.

References (cont'd)

- [Hammer78] Hammer M., McLeod D., "The Semantic Data Model: A Modelling Mechanism for Database Applications", ACM SIGMOD Conference, 1978.
- [Levesque79] Levesque H., Mylopoulos J., "A Procedural Semantics for Semantic Networks". in *Associative Networks: Representation and Use of Knowledge by Computers,* 93–120, 1979.
- [MacKay19] serc.carleton.edu/sp/library/conceptualmodels/index.html
- [Mylopoulos98] Mylopoulos J., "Information Modeling in the Time of the Revolution"*, Information Systems 23(3-4*), 1998.
- [Page18] Page S.*, The Model Thinker*, Basic Books, 2018.
- [Quillian68] Quillian R., "Semantic Memory", in Minsky M. (ed.) *Semantic Information Processing*, MIT Press,1968.
- [Ross77] Ross D., "Structured Analysis (SA): A Language for Communicating Ideas". *IEEE Transactions on Software Engineering 3(1)*, 16–34, 1977.
- [Roussopoulos75] Roussopoulos N and Mylopoulos J., "Using Semantic Networks for Database Management", First Very Large Databases Conference, Austin, 1975.
- [SEP05] "Concepts", *Stanford Encyclopedia of Philosophy*, 2005.
- [SEP18] "Associationist Theories of Thought", *Stanford Encyclopedia of Philosophy*, 2018.
- [Simon69] Simon H.*, The Sciences of the Artificial*, MIT Press 1969.
- [Yu97] Yu E., "Towards Modelling and Reasoning Support for Early Phase Requirements Engineering", 3rd IEEE International Symposium on Requirements Engineering, Annapolis, January 1997.