ELSEVIER

Artificial Intelligence in Engineering 15 (2001) 83–92

ARTIFICIAL INTELLIGENCE in ENGINEERING

www.elsevier.com/locate/aieng

Phenomenologial foundations of conceptual product modelling in architecture, engineering and construction

Žiga Turk*

Faculty of Civil and Geodetic Engineering, University of Ljubljana, Jamova 2, 1000 Ljubljana, Slovenia

Abstract

The ultimate goal of conceptual modelling in architecture, engineering and construction (AEC) has been to define the data structures that could be used to describe the entire built environment through all its life cycle stages — from inception and design to demolition. In spite of the magnitude and complexity of this task, the theoretical foundations of modelling received little attention. In this paper, the theoretical foundations of the traditional modelling approaches are questioned using phenomenology and hermeneutics as philosophical base. The author exposes the difference between the remodelling of some existing models, the modelling of physical objects and the modelling of psychical, intentional objects. The author concludes that AEC or building product and process models do not model objective reality but the modeller's partial understanding of that reality. Therefore, several correct but different models may and should exist. Future software architectures in AEC should not be built on a unified, centralized model but, on a combination of models, which may not be standardised but whose schemas are encoded in a standard manner.

Keywords: Product modelling; Conceptual modelling; Computer integrated construction; Phenomenology; Hermeneutics

1. Introduction

1.1. Modelling for computer integrated construction

Since late 1960s, it has been argued [1] that the lack of a standardised scheme for the exchange of information hinders the use of computers in architecture, engineering and construction (AEC). The exchange of information between many applications, as shown in Fig. 1 (left), was considered impractical. A standard scheme, shared by many applications, should be developed so that they could automatically exchange structured information about being designed, planned for, or building AEC products. Standardised data formats based on standard data structures have been considered the key enabling technology to achieve computer-integrated construction (CIC), as shown in Fig. 1 (right).

Conceptual modelling techniques were used since the mid 1980s to define conceptual models of construction products and processes (overview in Ref. [2]). An important part of research projects, such as COMBINE [3], CIMSTEEL [4] and ToCEE [5], funded by the European Commission, was a conceptual product model. Standardisation processes started in the ISO-STEP [6] and IAI-IFC [7] contexts. The development of these standards was rather slow; the adoption by the software vendors and AEC industry is even slower. As late as in the year 2000, the first IFC versions of CAD software became available that implements standard exchange formats for geometrical data.

1.2. About the paper

The author believes that the reasons for the moderate success of conceptual product modelling approach are rooted in the naive philosophical assumptions that were taken for granted by the AEC product modelling community and almost never seriously studied. The problems that the product modelling is facing cannot be solved by a new modelling language, a friendlier tool, a better organisation of the standardisation body or by a directive that would enforce the use of a 'standard' product model.

Instead, the complex philosophical and metaphysical background should be well understood. In this paper, the traditional, naive, background will be presented and compared to a more complex view offered by phenomenology. For example, in the naive approach one never really questions how the walls exist. Walls, beams, columns, etc. are surrounded by air. One could argue that all this is just a mixture of protons, electrons and neutrons. And yet we decided to separate, cut some of those particles and label them 'wall'. We take the 'being' of walls for granted,

* Tel.: +386-61-1768-622; fax: +386-61-1250-693.
E-mail address: ziga.turk@fgg.uni-lj.si (Ž. Turk).

0954-1810/01/$ - see front matter
PII: S0954-1810(01)00008-5

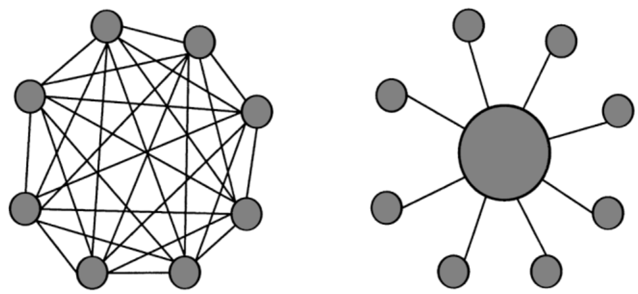

Fig. 1. Information exchange among AEC software — a motivation for product modelling.

without questioning what it means to 'be' or how walls came to be; not how walls developed through the history of architecture and building technology, but how we invented that they are a separate entity worth considering. The modelling community immediately jumps to define the attributes of walls, such as width, height, thickness, and structures the information about walls into more elaborate schemas.

However, the surprise and curiosity that things are, that they have being, is considered the origin of philosophy itself. Phenomenology is a branch of philosophy that deals with how to take things for what they really are [8] and what it means 'to be' in the first place [9]. The paper also touches on the subject of how humans act in the world and how this is fundamentally different to machines. Some argue that the human mind does not primarily operate rationally, on the 'mind models' we construct of the real world around us. Humans function well without (mind) models. Engineering software relies heavily on the models, but after reaching a certain level of bureaucratic intelligence and efficiency, the models can become an obstacle.

The philosophy presented is, of course, arguable, however it places the traditional assumptions of modelling into a broader context and contributes to a deeper understanding of how humans, like engineers and architects, work and think, and what computer software, implementing models, can do to help them.

1.3. Paper structure

In Section 2 the author defines the context of the paper — product and process modelling and presents the traditional understanding of the subject. In Section 3 he presents the alternative philosophical background. This is then used to explain and analyse current product modelling practise. The author argues that product models are subjective, cause blindness, harness creativity and are of limited use.

2. Modelling

The term *model* has many meanings including 'a usually miniature representation of something', 'a description or analogy used to help visualize something' and 'an example for imitation or emulation' [10]. It is an abstraction, simplification or idealization of reality that can be used for realization, simulation or as a prototype [11]. A more cautious term instead of 'reality' is 'original' or 'universe of discourse' (UoD) because it does not imply anything real or tangible. The process during which models are created is called *modelling*. A person making a model is a *modeller*. A *formal model* is a model that is encoded in a proper form, usually using mathematical or logical language.

2.1. Building product models[1]

It is generally believed that in order to be useful, computers must have knowledge of the domain in which they operate. In programs we define *digital models* of the original. Since computer programs are created to be useful for more than one original, the *schema of the model* and the model itself are separated. During systems analysis, programmers try to think of every possible original that the program will encounter and define a schema into which any original could fit. Instead of thinking about particular instances, they think about concepts common to all instances. We can, therefore, distinguish between (1) *conceptual modelling* that results in a *conceptual model* (source for a database schema), and (2) *engineering modelling* that results in the *data model* about a (designed) product. Note that conceptual model, in this context, has nothing to do with conceptual design. Conceptual designing is one of the early stages of designing that results in a rough design of a to-be-built product. This design can be described in the data model, just as a detailed design would be.

One can also distinguish between abstract and concrete models. By definition, all models 'abstract' reality, therefore the term *abstract model* can be used to denote models of abstract things (e.g. a tower) as opposed to a *concrete model*, which describes a concrete thing (e.g. the Eiffel Tower).

Product and process models model products and processes, such as 'the Eiffel tower' or 'construction of the new Boston tunnel'. *Conceptual building product models* define concepts about buildings and define schemata into which data about any building would fit. Data that fits the predefined building conceptual schema or structure is called the *building product data model*. In this paper, we use the term 'building product model' to denote the conceptual model, and not the data model. Product data models contain *structured data*. Building products can also be described by data that do not fit a predefined structure (e.g. a GIF image). This data, since handled by a computer, fits some structure, but not a specialised building structure. It will be referred to as *unstructured building data*.

Realizing the size and complexity of conceptual building

[1] 'Building product models' is a term by which the modelling community denotes product models in AEC. Terms 'AEC' and 'building' are used as synonyms in this paper.

product models, the developers looked at various mechanisms to structure them into manageable pieces. *Reference models* would prescribe a certain optics through which we should be looking at the world. In the GARM model [12] one should look at products as a hierarchy of functional units, each having several possible technical solutions. The form–function–behaviour model [13] would suggest to observe the products as each component having a form and position in the whole structure, the intended function would behave in a certain way under the influence of other parts or the environment. *Framework models* would suggest a certain breakdown of the products, for example, the RATAS model [14] is broken down into building, system, subsystem, part and detail. *Resource models* would provide useful building blocks for more complex models, for example, geometrical data.

2.2. Objects, symbols and concepts

An essential question at this point is 'What is it that is being modelled?' It was touched upon briefly in the definition of the term 'modelling'. There, the term 'real world' was cautiously replaced by 'original' or 'universe of discourse'. In the conceptual modelling practice, however, little distinction is made. In building product modelling, real world objects, such as walls, beams or columns, are being modelled and the 'reality' of these things is hardly questioned.

According to the traditional rationalistic philosophy, the difference between reality and our understanding of that reality is not an issue, because it claims that there exists a rather simple mapping between the two. Rationalistic tradition believes that our ability to intelligently act in the world around us is due to the mental representations of the real world that we have in our minds. Sowa [15] refers to them as *mental models*. Tradition claims that we think in terms of objects and their properties that are (more or less faithful) models of objective reality. For example, if an engineer says, "this beam is cracked", the terms 'beam' and attribute 'cracked' apply to some objectively present beam that is objectively cracked. Knowledge is storage of such representations. When we need to think about a problem, we retrieve a relevant representation, or a frame of related representations [16] and adapt it to the current situation.

This tradition can be traced back to Aristotle, who introduced a three-way distinction between *words*, *experiences in the psyche*, and *things*. Early in this century, Ogden and Richards [17] proposed a similar schema in the form of the Ogden meaning triangle. It connects symbols, concepts and things (Fig. 2). Each of the points on the triangle indicates a separate component that may be involved in thought or communication. The object is any entity from some real or imaginary world about which an idea is held. The concept is the idea or thought of the object as held in the mind of a person. The symbol is an auditory, visual, or other form of utterance that is considered to stand for the object when communicated as part of a language.

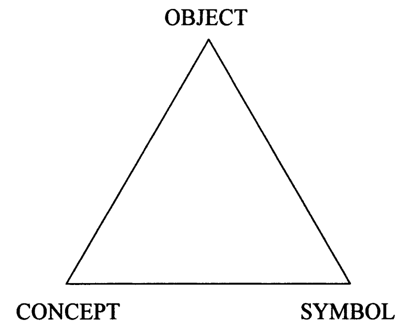

Fig. 2. Ogden's meaning triangle.

2.3. Mind, models and software

Such understanding of the human mind has provided the theoretical foundations for information and database systems development [15]. The ideas have also been adopted by the AI and cognitive science community [18] that placed great hopes in the possibilities that computers might become just as intelligent as humans are. How? It has been suggested that intelligent systems achieve their intelligence by manipulating symbols of the real world objects. Humans do it by manipulating 'experiences in the psyche', concepts or 'mind models'. Computers could manipulate symbolic representations of the models. Therefore, computers could achieve intelligence, just as humans do, if only the models and the manipulation rules could be complex enough.

Such understanding of our thought processes had a profound impact on computer programming. Typical (crude) steps in designing any computer program include:

1. Creating a schema of the universe of discourse — for example a model of finite elements that will represent the physical properties of a building structure.
2. Writing a formal representation of the model in the form of data structures and procedures — a finite element analysis program.

Typical use of such systems include:

1. Abstracting and mapping reality into concepts required by the database — creating a model of a structure based on geometric and material properties.
2. Writing the model in such a way that it populates the data structures designed by programmers.
3. Running the program.
4. Mapping the results back into reality.

Programming is, therefore, letting a computer know the concepts we have about the world. The symbols used in the computer (zeros and ones) will be different from what humans use to symbolise the concepts (sounds, words, graphical notation) but the theory claims that this is not important as long as the basic symbol manipulation mechanisms are preserved. Using programs is mapping real world objects into a computer's symbol and back. Such an approach is also the foundation for contemporary database designs and standards [6,19]. It is the foundation for the product modelling approach to computer integrated construction.

3. Phenomenology and hermeneutics

In philosophy, there is no consensus that the above relations between the real world objects, symbols that we used to talk about and concepts which we think about, are correct or complete. Particularly in the 20th century, the crucial role of the 'real world object' has been questioned, both in relation to the concepts as well the symbols. Finally, the mechanisms of thought that bind the three together were questioned as well.

In this section, phenomenology and hermeneutics are briefly introduced. They are relevant to the modelling: phenomenology particularly, because in constructing models, one should examine things what they truly are; hermeneutics, because models are ultimately interpreted and hermeneutics is the science of interpretation.

3.1. What exists

To act intelligently in a domain (the alternative theory), humans do not need a model of the domain. For example, although structural mechanics models were not existent in the Roman era, great domes have been built. Although we still do not have a construction process model, construction processes are carried out. People, who do not have a model of the dynamics of a hammer, wall and nail, can hammer the nail into the wall.

In other words, most of the time humans live in the world without breaking it up into objects with properties. So how do objects come into 'being'? When do we begin to notice a group of atoms as a distinct object? Is that object really there or is it just an invention of ours? Before attempting to model objects, one should have clear answer to these questions.

The existence of things has been a topic of philosophy since its beginning and non-traditional answers to this question did not appear with Heidegger. The roots of the two different views on the world are in the midst of the ancient thought between Paramenides's philosophy and Plato's metaphysics [20]. For Paramenides things that are present, exist. For Plato, and this is an essence of metaphysics, things also exist that are not present. He introduces the distinction between presence and existence. A common example in philosophy primers is that according to darkness. Paramenides defines it as the non-existence of light; therefore, darkness does not 'exist' in its own right, while for Plato darkness exists.

We can find exactly the same argument in the discussions that the building modelling community had about enclosures and spaces. Engineering community argued that walls, floors and ceilings exist because they are present and should, therefore, be represented in a schema. On the other hand, architects argued that spaces, in fact voids like a kitchen or a bathroom, exist as well and should be modelled as well. Indeed they are not present. They are void, but we can think of them and that should, according to metaphysics, suffice. They are constructed and found useful in the skill — the 'techne' — of the architects.

3.2. Objects and subjects

Another important contribution to this discussion comes from Descartes [21]. It was he, who introduced the word 'object' into philosophy. The term is widely used in engineering as well as in computer programming and conceptual modelling contexts. According to Webster [10], an object *is something material that may be perceived by the senses*, but also *something mental or physical toward which thought, feeling, or action is directed*. The world comes from the Latin term ob-iectum, meaning what is 'thrown in front of us' or what 'we have thrown in front of us'. The original meaning of the term has nothing to do with something that is just there, on its own and is independent of ourselves or of others, just waiting to be referenced. On the contrary, we, the observers, or someone else has to invent it and throw it in front of us. The etymological meaning of the adjective 'objectively' is therefore just the opposite of what we use it for.

A related term is 'subject' — sub-iectum means what is thrown under, implying a self-sufficient, self-standing base. The scholastic philosophy considered all item subjects. Descartes introduced the distinctions between sub-iectum and ob-iectum, and raised only humans into the domain of subjects. Everything else is an object — something we choose to throw in front of us without a self-sufficient, self-standing base.

Objects are constructs, inventions not something that has existence independent of ourselves.

3.3. Intentional objects

Brentano [22] introduced the distinction between physical and psychical phenomena. The psychical ones include an intentional object. An intentional object is one that is inside a thought — with which we intend to do something. There can be as many intentional objects as we may have intentions. Intentional objects, however, need not be made up. They may (but do not have to) relate to an external or physical phenomena. Brentano defines the latter as phenomena that do not include an intentional object.

Brentano is considered one of the starters of phenomenology — a branch of philosophy that examines phenomena for what they essentially are. Its best-known representative is Martin Heidegger.

3.4. Hermeneutics

Hermeneutics is the science of interpretation. Originally, the word was associated with the interpretation of biblical texts. In the 19th century, methodological hermeneutics tried to develop systematic procedures that would lead to correct interpretations of texts by studying the context, in which they were written. In the 20th century, based on the works of his teacher Heidegger, Gadamer [23] developed philosophical hermeneutics. It was not so much concerned with the intended meaning of the text, but with the acting upon a text in the reader's current context. Later Ricoeur suggested phenomenological hermeneutics and claimed that meaning is construed by the reader, not just according to the author's context but also according to its significance in the reader's context. Meaning is always achieved at the receiving end and if phenomenologically released from the writer's intent, several plausible interpretations of a text are possible.

4. Synthesis

Hermeneutics and phenomenology provide a somehow different perspective on how humans function. How we function should be understood if computers are to assist us:

- *Concepts*. Phenomenology takes a closer look at how to find out what the real world objects really are and seems to find little distinction between real world objects and concepts. Moreover, it finds conceptual representations of the real world rather insignificant.
- *Symbols*. Communication is not about transferring information about the real world, using symbolic representations of the real world objects, but simply means to get the listener to (re)act to the speaker's message so that people can negotiate the commitments among them. The speech act theory of Austin and Searle [24] elaborates on this.
- *Mechanisms*. Problem solving is not a search in a space of potential solutions. The essence of intelligence is finding a solution outside the predefined search space [25].

The last two items will be elaborated in this section. The first one was discussed in the previous one.

4.1. Objects in thinking

Hermeneutic phenomenology has mostly been influenced by Heidegger. At the core of his philosophy is the understanding that the basis for our everyday action is the ability to act pre-reflectively when 'thrown' into a situation — without reflecting or thinking when forced to react in a certain situation. He claims that reflective thought about objects and properties is derived from pre-conscious experience of them as 'ready-to-hand'. The essence of intelligence is 'throwness', not reflection. This is contrary to the tradition, which would consider reflective analysis of a detached observer as the basic intelligent behaviour.

To hammer a nail, Heidegger argues, we do not require conscious reflective knowledge about the physical properties of a hammer and the physics of hammering. The tool is ready-to-hand and we just hammer the nail into the wall. Similar pre-reflective actions can also explain the so-called 'intuition', 'insight' or plain 'common sense' that are sometimes used by the designers or engineers to explain their creative process. A good idea just pops out of nowhere, often while we managed to get out of a theory — while walking a dog or riding a bus. The solution is not a result of a systematic search in the set of potential solutions. *The essence of intelligence is to act appropriately when there is no simple pre-definition of the problem or the space of states in which to search for a solution* [26]. Studying how professionals think in action, Schön [25] came to similar conclusions.

Heidegger does not claim that reflective action does not take place. On the contrary — in the case of problems and difficulties we begin to examine the world in terms of objects and properties (Fig. 3). He uses the term 'breakdown' (not in the structural mechanics sense) for such situations. The beliefs of the psychologist Frankl [27] confirm this. He claims that the main driving force of humans is the *Wille zum Sinn* — the will to meaning. We want to make sense of things, particularly when they go wrong, and find meaning in what is happening in our lives.

The models and theories developed after breakdowns have both positive and negative implications. They ensure that we avoid similar breakdowns the next time — and not only us, who defined the model or the theory — others as well. People have an ability to accept others' models and

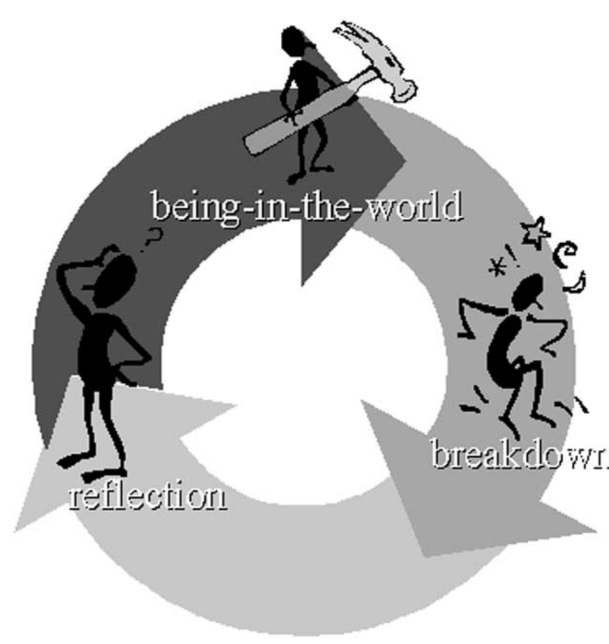

Fig. 3. The reflection loop.

theories. They can follow the recipes and solve problems with much smaller intellectual effort. It has been argued though, that things learned through models and theories are not as solid as the tacit knowledge learned through one's own experience [28].

4.2. Objects in communication

In a hermeneutic view, a statement "a wall is cracked" does not stand for some objective reality, but gets its meaning only in a certain context. The speaker knows that the audience expects a technical evaluation of a nearby structural element. On a finest level, every wall has some microscopic cracks so an objective statement that a wall has cracks is pointless. However, the speaker assumes how the audience will interpret the term 'cracked'.

The main ideas of hermeneutics and phenomenology related to the representation and 'information exchange' are:

- Language is not an exchange of information. It is not a medium for description but one of action. The statement about cracks in the beam, in fact, engages the audience to do something about the cracks. It creates a commitment to do something about the cracks.
- Language is a form of social behaviour that creates a 'mutual orientation'. By talking to each other, we establish common views and ideas. A consensual view on the world around us is not based on what that world objectively is, but what the speaker and the audience, collectively, consider it to be.
- The meaning of a statement is created at the receiving end and is not independent of it. Meaning is created within commitment networks. Commitment network is a social network of people who work or live together and give or accept commitments that they will do something.
- Understanding of a message can only be achieved, if the speaker and the audience share mutual concern and a common background. The unspoken is at least as important as the spoken part of a message.

4.3. Object based world

Physics tells us that the world is made of atoms and subatomic particles. Larger objects with properties are human inventions, human conventions, and do not exist outside human procedures; procedures that did break down. (In fact the same goes for atoms, but most of us do not have tacit experience of them, therefore, they were used as an illustration.)

A hammer does not exist as a separate object unless there is problem with it, i.e. unless there is a breakdown. While we hammer without problems, we do not experience the hammer as a special object. As objects and properties pop into existence only if there is a breakdown, things do not have properties independent of the interpretation and the context of the breakdown. It should be stressed that objects pop into existence, not just into thought. "Heidegger insists that it is meaningless to talk about the existence of objects and their properties in the absence of an activity of concern with the potential of breaking down" [26].

According to Heidegger, for example, a concrete beam does not have a modulus of elasticity (E). Concrete beams, for that matter, do not exist on their own. They are constructed out of a soup of atomic particles after a breakdown, i.e. after we have some problems with them. In Roman times, when they did not know about E, beams did not have that property. It became meaningful to talk about E only in the context of certain structural mechanic theories. In fact, E does not exist; beams do not even have length, beams do not exist. Length can only be established within a context where the users share a common understanding of the length of a beam (Fig. 4) — when we choose to throw such object or such property in front of us. This does not mean, of course, that there is no certain amount of concrete in a certain location in the universe. (Although some philosophers claimed that the greatest caveat of modern science is that it cannot prove that the real world actually exists; that it is really out there, and not just an illusion. But this discussion is not relevant to our discourse).

5. Phenomenological view on modelling

In building product modelling, we model phenomena. But the question is: which ones? the physical or the psychical? Should we really care about the distinction between them?

5.1. Models of models

In engineering and construction, the term model was first associated with structural, mechanical models, which seem physical enough, but the question was hardly ever revisited, when modelling was used to capture large scale design and business processes.

Conceptual models for systems analysis may model an existing mathematical, structural or mechanical model. In the case of the creation of a model of a model, e.g. while creating a data structure for a finite element program, a clear, concise definition of the 'thing modelled' exists and the philosophical issues do not arise. The finite element theory, for example, defines that to calculate the deflection of the beam; we must know its geometry and modulus of elasticity and perhaps a Poisson's number. Conceptual modelling must define a schema into which such data fits.

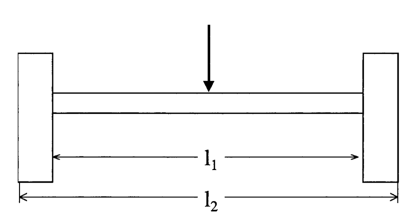

Fig. 4. Lengths of a beam.

The meaning of this data is defined within the theory of elasticity. The objects modelled are not real world objects, but constructs that are given meaning within some theory.

5.2. Physical models

Do building product models model something physical? The naive answer is yes, because, yes, walls seem physical. But are they?

Building product models are definitively not models of models and go beyond a single theory. They are defined as the 'totality of all information about a building product throughout its entire life cycle' [14]. From a software developer's perspective, this definition would imply a union of all schemata of every program that can ever be written for a construction application.

From a philosophical perspective, a building product model would imply a model of all possible intentional objects that we may invent to think about a future 'physical' object — a building or a bridge. How a physical object is broken into intentional objects depends on the intentions we have with it, and these are impossible to predict in the process of modelling.

The product models are not being defined bottom up — as a union of schemata of existing software — but top down using a naive phenomenological approach. Let us look at what a beam 'really' is and define the data structures we need to describe it. Instead of providing schemata for the data required within a self-contained, well-defined universe, conceptual product modelling is trying to define the schemata for the intentional objects that are in our minds only or may be thrown in front of us (ob-iectum!) in the future. Needless to say, it is not possible to model things that we will only think about in the future.

5.3. Intentional models

We can conclude that building product models are in fact *intentional models* of anthropo-technical nature. Their nature is technical because they are not associated with the nature itself, but with the 'techne' — the skills, which humans use to reshape the nature. They are anthropo-technical (and not socio) because they are based on an individual's understanding and intention.

6. Caveats of intentional models

The intentional nature of building product models is the source of most problems the building product modelling community is facing.

6.1. Vicious circle of modelling

Intentional models are vulnerable and unstable. A conceptual, intentional, anthropo-technical model can change one's understanding of the concepts themselves. The author calls this the *vicious circle of modelling*: 'Reality' of anthropo-technical topic is a matter of convention and not of an objectively present reality: a group agrees what really is. For example, does a 'resource' include or exclude people. Models influence the mutual understanding of a topic: a model of a resource makes people rethink or adapt to what they consider it to be. So the model changes the universe that was observed and modelled. It should be stressed again that in this case, the universe has been our own understanding of the topic, what we chose to ob-iect in front of us, and not some self-standing real world, independent of us. Reference and framework models are particularly dangerous in distorting our perspective on the universe of discourse.

6.2. Stimulating some intentions — fixation

Design fixation is a phenomena experienced by designers when their previous designs do not allow them to escape the examples or patterns of those designs when trying to create something new and original [29,30]. At first glance, conceptual product models are not previous designs. They are, however, models of intentional objects that the conceptual modeller expected to be useful as design objects if instantiated. Instantiation is not a discrete, but a continuous process. Designing conceptual product models for software is one step in this gradual instantiation of the conceptual model into data model. In this perspective, conceptual product models are designs as well and a designer might well be fixated to the solutions suggested by the conceptual model. For example, if a conceptual product model only includes orthogonal windows and doors, designers would be locked onto making all doors and windows orthogonal.

6.3. De-stimulating intentions — blindness

The very instant we begin to think reflectively about a situation and analyse it in terms of objects and properties, we disconnect ourselves from the *being-in-the-world*, we are not *thrown* into a situation any more. We limit our view of the problem to the one that can be expressed by the objects and properties we have adopted, and become 'blind' for all other possible solutions.

For example, in the bridge selection expert system, only the structural designs built into the expert system can be 'designed'. Even human engineers, while looking at the problems with a closed mind, show symptoms of blindness. It has been argued [31] that the collapse of the Tacoma Narrows Bridge was due to the blindness of the designers in the knowledge of aerodynamics.

Pre-defined conceptual product models create blindness for design options that cannot be expressed using the objects and relationships of the product model. The fact is a bit more obscure, but nevertheless still present, when designs are generated by combinations, through structure evolution or exploration [32].

Blindness and fixation have synergetic effects. While the first obscures solutions that cannot be represented by the

Table 1
Comparison of technologies

	Expert systems	Product models
Problems in the software development phase	Difficulty to extract expert knowledge from the experts	Difficulties to define and agree upon models required by a spectrum of applications
Problem during the use of the software	Blindness is created by excluding large portions of the background knowledge	Models restrict the designs and/or solutions to those that can be represented in a model. A blindness for alternatives is created

conceptual models, the second promotes ones that can be represented by a model.

Some design theorists are aware of these limitations [33], but their influence on the mainstream conceptual modelling in building has been small.

7. A phenomenological view on product modelling

In this section, a phenomenological view on the open issues of the conceptual modelling in building is given.

7.1. Explaining problems

There are several issues related to product modelling that can be explained with phenomenology:

- In most research projects in AEC, researchers chose to define their own conceptual product models. In the light of the theories presented, it could be argued that this is so because conceptual models are subjective, and it is easier to model one's own understanding than adopting someone else's. We described in Section 4.1 that humans reflect and create models when they face a problem. Therefore, it is very natural that when faced with a problem in the field of computer-integrated construction, a model is created.
- Defining a common conceptual model, even within a small research team, is difficult and leads to endless discussions. If conceptual models were objective and would be based on some hard scientific facts about the domain, this could be used as an argument in the discussion. But, usually they are not; there is no reality out there to check the models against.
- There is no way to prove a model correct or wrong, better or worse, except for its internal coherence. All one can say is that it does or does not meet some rather vague requirements. This is similar to the evaluation of the works of an architect — among the proposals that meet the requirements, a committee select the most 'beautiful' one. In engineering, conditions of satisfaction are defined much more unambiguously — meet the requirements at lowest cost. This makes conceptual modelling closer to an art than to a science.
- The conceptual product models in AEC are to an extent similar but essentially different from each other. In research papers and reports we read that they are correct, successful, applicable, etc. Usually, in science, several people come to the same result independently.

7.2. Extrapolation of research into industry

In the 1980s, expert systems were considered as a very promising information technology. Today, they are often used as an example for failure of a technology. What the product modelling community should worry about is that there are some similarities between the product modelling approach and expert system techniques (Table 1).

Both approaches believed that it is possible to create a computerised representation of the knowledge (expert systems) or information (product models) in a given domain. Nevertheless, it seems close to impossible to define complete models that would really make them useful in practice. In research prototypes, it has proven possible to define partial models or 80%-of-the-time-right expert systems. The work to extrapolate the research prototype into an industry strength application should not be underestimated. In the case of expert systems, the critics have argued that to cover the remaining 20%, an enormous amount of background and common sense knowledge would need to be computerized [34]. In the case of product models, the issues are similar. The modelling of data needed to support the activities for which theoretical or physical models exist is viable. However, the modelling of data that appears in activities that are handled in an ad hoc basis, using common sense, improvisation, etc. is not possible. Civil engineering is a domain where quite a few tasks are handled in such a way.

7.3. Product models are interpretations not representations

In Section 5, it was shown that product models are anthropo-technical models. They are not models of objective reality but of individual's intentions about that reality achieved in a given context. The exceptions are those parts of the product model that are defining a data model for a scientific theory. This theory then provides unambiguous definitions.

While developing standardised product model, a committee can achieve a common interpretation and later impose it on anyone using the standardised product model. As the interpretation is a function of the context, domain, breakdown, culture, etc., it is unlikely that the standardised model could match it all. To create complete models, the entire cultural context of humanity should be modelled.

7.4. Product models should leave room for interpretation

Unambiguity and strict definition of the semantics is usually considered as a positive feature of the product models. But in the light of the previous section ambiguity, or rather, possibility of different interpretations, is a very important feature of a product model because it lets the human compensate for the deficiency of the model. Usually, this can be achieved if the product model contains enough very generic components. For example, a 'line' can be interpreted as many things, including a wall. A wall, on the other hand, is just a wall.

The same goes for software that implements the product models. The most successful programs are not those that try to fully model the domain in which they operate, but those that implement the most generic concepts of the domain and leave plenty of room for modification and evolution; not necessarily by adding less generic data structures but by leaving the user the freedom of interpretation of the existing ones. For example, generic tools, such as AutoCAD or Word, are more popular than specialised ones, such as ArchiCAD or SGML editors.

7.5. Co-existence with unstructured data

In the early days of computing, when memory was precious, data was carefully structured. For example, years were represented with the last two digits only. Structured, indexed or classified data also helped the computers to be more efficient. Today, both RAM and disk space is cheap, and the CPUs are still doubling their speed every 18 months. What remains expensive is the human effort to structure that data, both during software development and during software use. Looking at the designers' hard disks, we can see the amount of the data created and stored increasing rapidly. The structured fraction is relatively smaller and smaller. We can assume that the structured part of building product data will constitute a smaller part of all data related to a building product. To be practical, product models must include or relate to the majority of unstructured data.

7.6. Product models for creative design: a contradiction of terms

Model-based software can support routine, bureaucratic processes that fit into the schema designed during software development. But can model-based software support creative design? In the light of hermeneutics and phenomenology, the term 'conceptual model for creative design' is a contradiction in terms. Only designs that fit into the model are possible. Creative ones, by definition, are those, which step out of the pre-defined conceptual model.

What has been seen as an answer are 'extensible' models created in such a way that the schema could be extended at run time [35], models that would support dynamic schema evolution [36], explorative models [32], etc. But does this really make any difference? So what if the user can add an attribute 'Poisson's number' to the schema. Of what use is it unless there is code in the finite elements program that implements it? In other words, it takes a computer programmer to use software that supports creative design.

This does not mean that computerised tools for creative design cannot be created. If we accept the idea that if being-in-the-world is important for the human mind, then we must make sure that software does not stand in the way. Therefore, pen and pencil are so popular with the architects and engineers in the initial stages of design because they are not in the way, they are not limiting the creative process.

8. Conclusions

Conceptual modelling has been analysed using phenomenology and hermeneutics as the philosophical background. Limits of conceptual modelling in building have been exposed. The conceptual modelling of anything but other models may lead to problems unless the modeller is aware of the intentional nature of the models:

- Conceptual models capture a part of relevant data for the building processes. The common sense, social context, general information in the social context should not be underestimated.
- Conceptual product models are subjective interpretations. This should be kept in mind while defining or discussing them. They are not right or wrong, but appeal to more or to fewer people.
- Models cause blindness and fixation. Creative work is least hindered by models that are so generic that they are not blinding or limiting.
- The success of the research prototypes is relative. Product model and expert system technology faces the explosion of background knowledge and the richness of possible contexts when tried in an industrial environment.

Based on this, some research directions for construction IT include the following:

- Coexistence of structured information and unstructured data; non-model-based technologies; technologies that rely on the human to interpret the data.
- Multiple models disclose the fundamental nature of anthropo-technical models. Several current architectures go in this direction: the different views on the same model [37] or different aspect models sharing the same core model. However, they are cautious of breaking the relations between different models altogether.
- Software that does not build a barrier to engineer's being

in the world but rather enhances it (like virtual reality, enhanced reality, multimedia).

- Communication software to support commitment negotiation instead of information exchange (Internet, groupware, collaboration, etc.).
- Software architectures, where standard, centralised conceptual models are replaced by a federation of proprietary models. The only thing these would have in common is that their private models are defined in a standard, machine readable way, for example using logic programming [38] or XML [39]. With the recent advances in schema and data encoding standards, the alternatives presented in Fig. 1 should be critically revisited. It might prove easier to define the many translators than to continue the quest for the schema of the dot in the centre.

References

[1] Champion D. A proposed unified schema for AEC information vs. transaction-centered multi-schemas. Architect Des 1967.

[2] Eastman CM. Building product models: computer environments supporting design and construction. Boca-Raton, FL: CRC Press, 1999.

[3] Augenbroe G, editor. COMBINE 1 Final Report. EU DG XII Joule Report.

[4] Watson A, Crowley A. Deploying CIMSTEEL integration standards. In: Competitive deployment of product data technology. Proceedings of the Third European Conference on Product Data Technology. p. 39–46.

[5] Turk Z, Scherer RJ, editors. ToCEE Esprit 20587 CD-ROM ISBN 961-6167-37-5Slovenia: University of Ljubljana, 2000.

[6] International Standards Organisation. ISO 10303, Industrial automation systems and integration — product data representation and exchange; 1994.

[7] International Alliance for Interoperability, http://iaiweb.lbl.gov/, accessed 15.1.2001.

[8] Sokolowski R. Introduction to phenomenology, ISBN: 0521667925. Cambridge: Cambridge University Press, 2000.

[9] Heidegger M. Being and time. New York: Harper & Row, 1962.

[10] Merriam Websters On-Line Dictionary, http://www.m-w.com/, accessed 15.1.2001.

[11] Petri CA. Modeling as a communication discipline in measuring. Modeling and evaluating computer systems. New York: North-Holland, 1977.

[12] Gielingh W. General AEC reference model (GARM). In: Christiansson P, Karlsson H, editors. Conceptual modeling of buildings. CIB Publication 126. Sweden: Lund University; 1988.

[13] Gero J. Design prototypes: a knowledge representation schema for design. AI Mag 1990;11(4):26–36.

[14] Björk B-C. A unified approach for modeling construction information. Building Environ 1992; 27(2):173–94.

[15] Sowa JF. Conceptual structures: information processing in mind and machine. Reading, MA: Addison-Wesley, 1984.

[16] Minsky M, editor. Semantic data models. Cambridge, MA: MIT Press, 1969.

[17] Ogden CK, Richards IA. The meaning of meaning. New York: Harcourt Brace Jovanovich, 1989 (First published 1923).

[18] Gardner HE. The mind's new science, ISBN 0-465-04635-5. New York: Basic Books, 1987.

[19] International Standard Organisation. ISO/IEC 10728, Information technology — information resource dictionary system (IRDS) services interface, 1993.

[20] Russell B. A history of western philosophy. 2nd ed. London: Routledge and Kegan Paul, 1991.

[21] Cottingham J, editor. Descartes: meditations on first philosophy ISBN: 0521558182. Cambridge: Cambridge University Press, 1996.

[22] Brentano F. The theory of categories. Hague: Martinus Nijhoff, 1981 (translated by Roderick M. Chisholm and Norbert Guterman).

[23] Gadamer H-G. Izbrani spisi [selected writings — in Slovenia], Ljubljana, Nova revija; 1999.

[24] Searle JR. The rediscovery of the mind, ISBN 026269154X. Cambridge, MA: MIT Press, 1994.

[25] Schön DA. The reflective practitioner: how professionals think in action. New York: Basic Books, 1990.

[26] Winograd T, Flores F. Understanding computers and cognition. Reading, MA: Addison-Wesley, 1997.

[27] Frankl V. The will to meaning. New American Library Trade, 1989.

[28] Schön DA. Educating the reflective practitioner: toward a new design for teaching and learning in the professions. San Francisco, CA: Jossey-Bass, 1990.

[29] Jansson DG, Smith SM. Design fixation. Des Stud 1991;12:3–11.

[30] Purcell T, Williams P, Gero JS, Colbron B. Fixation effects: do they exist in design problem solving? Environ Plan B 1993;20:333–45.

31 Ferguson ES. Engineering and the mind's eye. Cambridge, MA: MIT Press, 1992.

[32] Smithers T, Troxell W. Design is intelligent behaviour, but what's the formalism? AI EDAM 1990;4(2):89–98.

[33] Snodgrass A, Coyne R. Is designing hermeneutical? Architect Theory Rev, J Dept Architect, Univ Sydney 1997;1(1):65–97.

[34] Dreyfus HL, Dreyfus SE. Making a mind versus modeling the brain: artificial intelligence back at branchpoint. Daedalus 1988;177(1):185–97.

[35] Eastman CM, Assal H, Jeng TS. Structure of a product database supporting model evolution. In: Proceedings of the CIB W78 Symposium, Stanford, CA. 1995.

[36] Ekholm A, Fridqvist S. The BAS * CAAD information system for design principles, implementation, and a design scenario. In: Proceedings of the Eighth International Conference on Computer Aided Architectural Design Futures [ISBN 0-7923-8536-5], Atlanta, 1999 June 7–8; p. 149–64.

[37] Rosenman MA, Gero JS. Modeling multiple views of design objects in a collaborative CAD environment. CAD 1996; 28(3):207–16 (Special Issue on AI in Design).

[38] Turk Z. Semantic representation of product modeling data, Pre-Proceedings, MITC Conference. Singapore: National University of Singapore, 1993.

[39] W3C Consortium. XML Protocol Activity, http://www.w3.org/2000/xp/, accessed 15.1.2001.

***Žiga Turk**, 1962, holds a B.Sc. and a Ph.D. in structural engineering and an M.Sc. in computer science from the University of Ljubljana, Slovenia. As an associated professor, he lectures in Ljubljana, was a visiting lecturer at the KTH in Stockholm and is a visiting professor at the ITU in Istanbul. His research topic is construction information technology, particularly computer-integrated construction, engineering information systems and communication technology, such as the Internet where he first set up construction related services in 1993. Internationally, he has published over 60 conference and journal papers. Dr. Turk is involved in editorial or scientific boards of two journals and in the committees of many relevant international associations, such as CIB W78, IABSE's WC6 and ECCPM. Dr. Turk's homepage is at http://itc.fgg.uni-lj.si/~zturk/.*