Chapter 13. Modelling

Table of Contents

What is a model?

A model captures the essential aspects of a system or process while suppressing details. It should be clear and concise. It need not cover all aspects of that system or process, but provide different views of it. Examples in IT are the relational model of data, the ANSI/SPARC Database Architecture, the waterfall model in software engineering, the ISO OSI model for network protocols, etc. A model can be used as a blueprint for developing new systems/processes, or for studying, evaluating and improving existing ones.

Models, like prototypes, make it easier to reason about ideas and communicate them more clearly to others. A model deals with the conceptual behaviour and function of a system whereas a prototype is concerned with lower-level details such as structure, components, methods and operations. A model is a simplified, abstract representation of an idea/design; while a prototype is a simplified, concrete implementation of (aspects of) an idea/design.

Advantages of models

A model can be used early in your research to help you design a system, case study, process or experiment, or to formulate a hypothesis. It can also be used at the end of your research to clarify what the product of your research is, or to prove some properties it exhibits.

It is commonly accepted that humans can only keep seven (plus or minus two) concepts in mind at a time, and hence the simplification provided by a good model facilitates understanding and reasoning about a complex system/process. Models can address all aspects of a problem in different modeling views, because detail is hidden in each view and only relevant aspects highlighted. This is useful for implementers to ensure that they comprehensively cover all the necessary issues and facets of the system/process.

Studying a model can lead to insights that would be harder to gain from consideration of the system/process itself — e.g. if the model is based on a real-world analogy which is familiar and well understood, then new ideas often arise from considering how the real-world metaphor is used. The formal semantics of the model clarifies the system/process components, flows and requirements compared with other types of system/process descriptions which are often incomplete, ambiguous or unnecessarily constraining.

Modelling lifecycle

When a researcher realizes or suspects that s/he has discovered a new problem, a model can help to express this clearly and concisely and in so doing clarify the problem (and confirm or deny its existence). This model can be studied to determine what assumptions it is based on, and then alternative models devised based on different assumptions. The assumptions underlying these models are made explicit, and the models compared and the implications of different assumptions investigated. Once several models for the system/process have been proposed, more general models can be developed

that cater for most or all of these initial models. This typically is repeated when new models, not covered by the general model, are proposed and the general model extended to cater for them, and this is possibly repeated several times over.

Proposing a new model

If your primary research aim is to propose a model, you must start by making sure that you are familiar with all existing models for this situation/problem. If models already exist, you will need to show how yours differs and why it is needed (is it simpler? applicable in some cases where existing models are inadequate? more precise and thus more suitable for analysis?)

The more models already exist, the less interesting it becomes to propose a new one and the more important it is to show significant improvements of using the new model. One way is to show that it is more general — it can support everything provided by existing models (and possibly still more that is not supported by any known model). Simplicity is desirable in a model, a model that can be shown to be simpler than existing models and as expressive (or more) can be argued to be better than existing models.

Sometimes a new paradigm is introduced in one field of Computer Science (e.g. AI) which you believe might be usable in another field (e.g. databases), so you can attempt to create such a model for using the new concept in this other field. If this has already been done, you could consider different aspects within the other field (e.g. query processing, concurrency control, distribution, etc.) and find one which requires a new model if the new paradigm is used.

It is sometimes unclear if a new problem, process or paradigm requires a new model. If you manage to propose a new model which reveals some interesting new facet of the problem/process/paradigm, or offers some benefit in that context, then your work will be interesting and useful to the research community.

How to devise a new model

A model can be derived from a prototype design and/or implementation of the system you wish to model — capturing its components, flows and essential properties after completing your design or implementation. Another way to develop a model is to find a suitable real-world analogy (just think e.g. of such well-known models as windows, files, folders, objects, agents, etc.). Using a real-world analog can often give you greater insight into the system you are modeling, but thinking of the components and processes typically associated with the real-world metaphor. For this approach, you need to know what you want to model well enough to understand its essence, to think creatively and to look at your system with an open mind. Alternatively, you can use well known formal models such as logic, sets, formal languages, automata theory and the like.