

How to faithfully model systems composed of millions of parts?



Knowledge Inside
architect your system

Samuel Boutin (Knowledge Inside) - sbo@k-inside.com

Copyright © 2022 by Samuel Boutin. Published and used by INCOSE with permission

Keywords.

complexity
system model
patterns in systems
viewpoints
diagramming complex systems

Topics. 1.1. Complexity; 1.3. Natural Systems; 1.6. Systems Thinking; 2.4. System Architecture/Design Definition; 3. Automotive; 3.3. Decision Analysis and/or Decision Management; 3.6. Measurement and Metrics; 5.3. MBSE & Digital Engineering; 5.4. Modeling/Simulation/Analysis; 8. Energy (renewable, nuclear, etc.);

Abstract. Our presentation is about faithfully modeling complex systems, whether they are human made or not. In particular, we explain how to handle modeling of highly repetitive systems and try to specify what is a faithful model in this context. By highly repetitive, we mean systems that may include several hierarchical levels and millions to billions of parts organized according to a small number of patterns. Such a complexity raises the question of relevant intentional versus extensional modeling. The usual flaws are that the extensional representation can be too complex while the intentional representation can be too abstract: we think we found an elegant mix.

Many systems include a high number of parts or subsystems. This is the case for natural systems: if you want to model a lung with its bronchial system and all pulmonary alveoli, you will have to handle several hundreds of millions of similar objects. More generally, human body is made of about 10 000 billions of similar cells. It is the same for many human made systems around us: batteries can integrate several dozens of thousands of cells or even much more, a nuclear plant embeds several steam generators and thousands of pumps and pipes, a railway network is made of a very limited number of components and you may integrate hundreds of seats in a plane. Integrated circuits also embed a very repetitive and systematic design. Very often, we integrate redundant functions and components as a safety mechanism leading to parts replication. All systems made of thousands of parts (or much more) are usually made of a much smaller number of different parts.

In a case of systems containing many similar subsystems, similar does not mean equal, e.g. the “position” of “objects” may impact their interfaces and properties. When looking at a pack of cells, cells from “the border” will not have the same interface as cells from “the inside”. This is what we could call side effects. Managing such side effects is an important aspect to maintain modeling accuracy. In face of high complexity, Lewis Carroll and Borges tell us that a map on a one-to-one scale is useless. Repetition seems a useful lever but side effects need to be taken

into account. And despite the complexity we would like to get a faithful model. What a faithful system model should be deserves an explanation. Here are the properties we would like to keep for our modeling:

- Any parent-child relation between a system and its subsystems in the actual system shall be faithfully represented by a parent child relation between respective modelled subsystems in the modelled system.
- Any interface between a system and other systems in the actual system shall be represented faithfully by a relation between respective modelled systems.
- Any attribute of an item in the real system we wish to consider shall be represented faithfully by an attribute of that modelled item.

As an example, we will present a way of modeling a battery from cell level to battery pack level inside a single diagram. We also discuss how this system modeling approach fits with requirements management, functional architecture and behavior modeling. We estimate the complexity leverage of the presented approach by estimating the number of terms of the model w.r.t the number of parts of the real system.

Shift performance expectations in Systems and Projects engineering with arKItect



Knowledge Inside
arKItect your system