Exploration of Rosen’s Modelling Relation and Category Theory

Bruce McNaughton
Program Manager / System Architect

19-October-2024

An initial exploration. Any feedback or thoughts appreciated.

bruce.mcnaughton@change-aide.com

Rosen’s Modelling Relation (4 References).

Figure 2.3.1: Anticipatory Systems, Edition 1
Robert Rosen

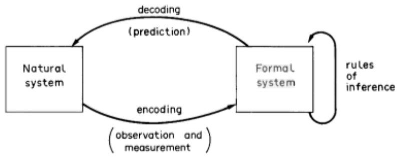
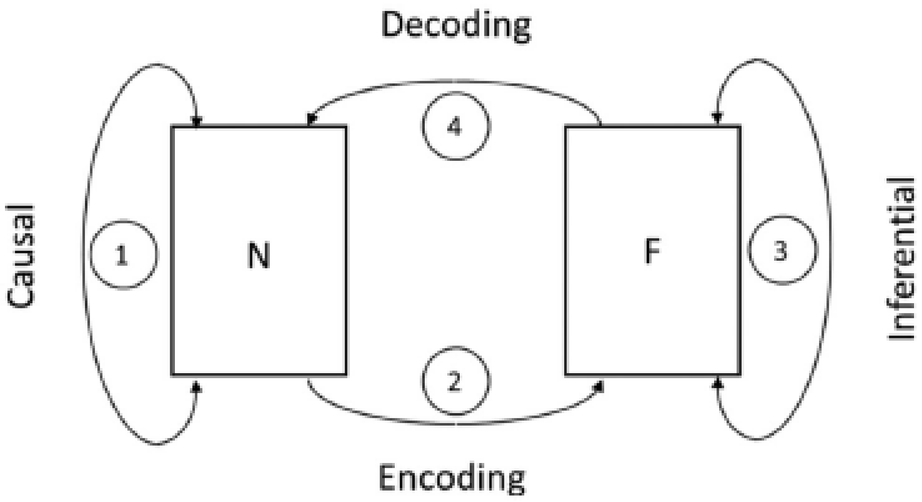


Fig. 2.3.1.

FIGURE 1 **Rosen's modelling relation**
From: Relational holon science and Popper's 3 worlds in engineering practice
David Blockley | Gary Smith | Patrick Godfrey | John J. Kineman
Blue Text Added.

Functor
Prediction based on model or simulation



Functor
Observation and Measurement

Category:
N = Natural System

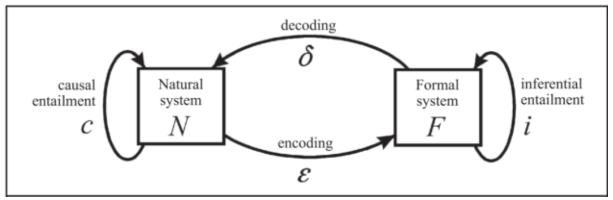
Category:
F = Formal System
Model of N

On Models and Modeling
Robert Rosen

Formal models were typically Mathematical models. Ensuring The results of the simulation matched The observations confirmed validity of The model conforming to the natural System.

FIGURE 2 **Prototypical Modeling Relation**
From: Robert Rosen’s Anticipatory Systems
A. H. Louie, 2010

Figure 2 The prototypical modeling relation



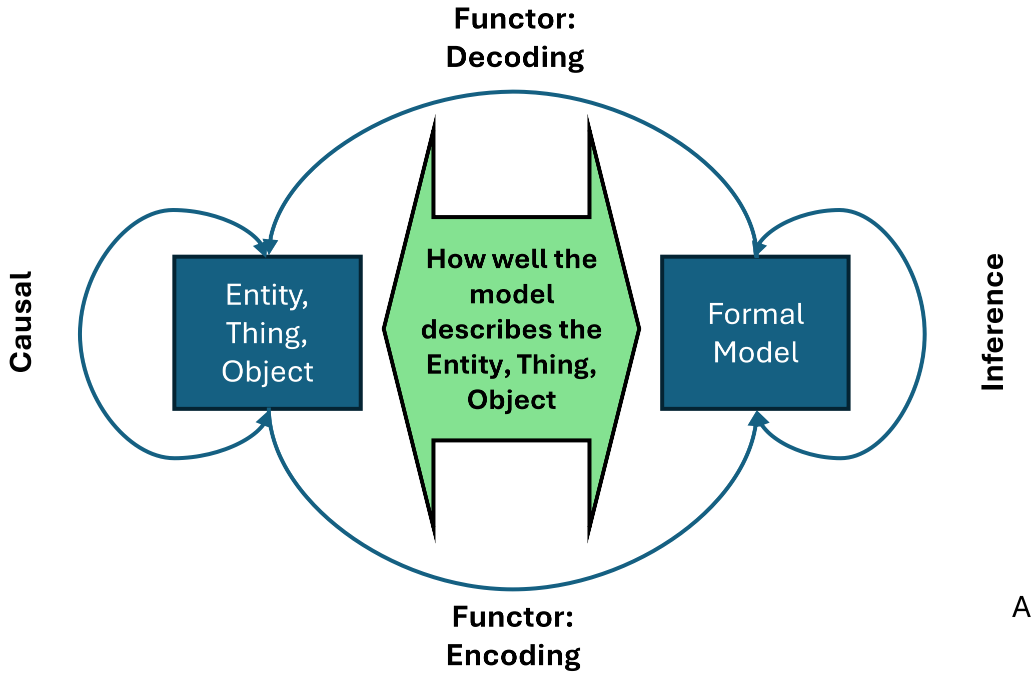
Using Rosen’s model

- **F = Formal System Model**
  - Formal system model is a conceptual model (or aligned mathematical model) of a Natural System.
  - The Formal System Model has a relationship with the Natural System based upon Functors
  - The Formal System Model has elements and relationships that can be mapped to the Entity, Thing or Object as morphisms (e.g. homomorphism or isomorphism, etc) in the Functors.
  - Relates to Epistemology
- **N = Natural System**
  - Natural System is an entity, thing, object in the natural world
  - Through observations and experiments there is a functor that establishes a relationship between the Natural System and the Formal System Model.
  - The observations and experiments can help create or improve a Formal Model.
  - Relates to Ontology
- **Two Categories (F, N) and Two Functors (Encoding, Decoding)**
  - This is described in Rosen’s book “Anticipatory Systems”, Edition 1 and the paper on models and modelling highlights the problems of misalignment of N as an entity, thing, object and the models F.

Rosen Model: Adapted for conversation

Rather than use the term “Natural System”,
The focus shifts to a physical or abstract:
Entity, Thing or object.

Has named properties
or characteristics



Focus on any type of model
About an Entity, Thing or Object

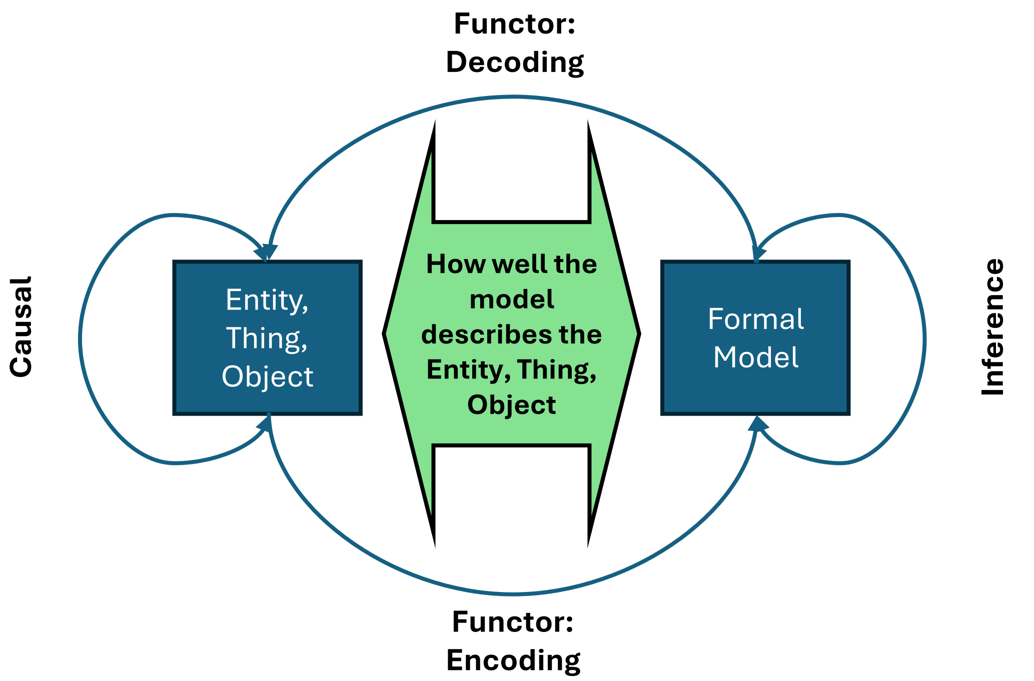
A focus on Ontology

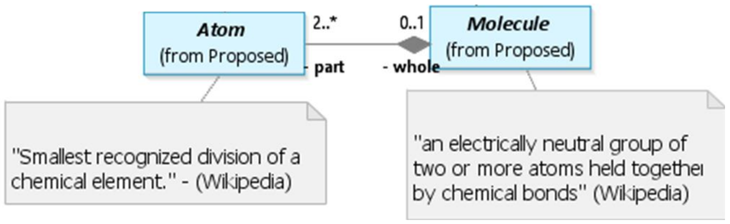
A focus on Epistemology

Rosen Model: Water Molecule Example

Entity, Thing, Object:
Water Molecule:
H_2O

The Entity, Thing, Object is H_2O.





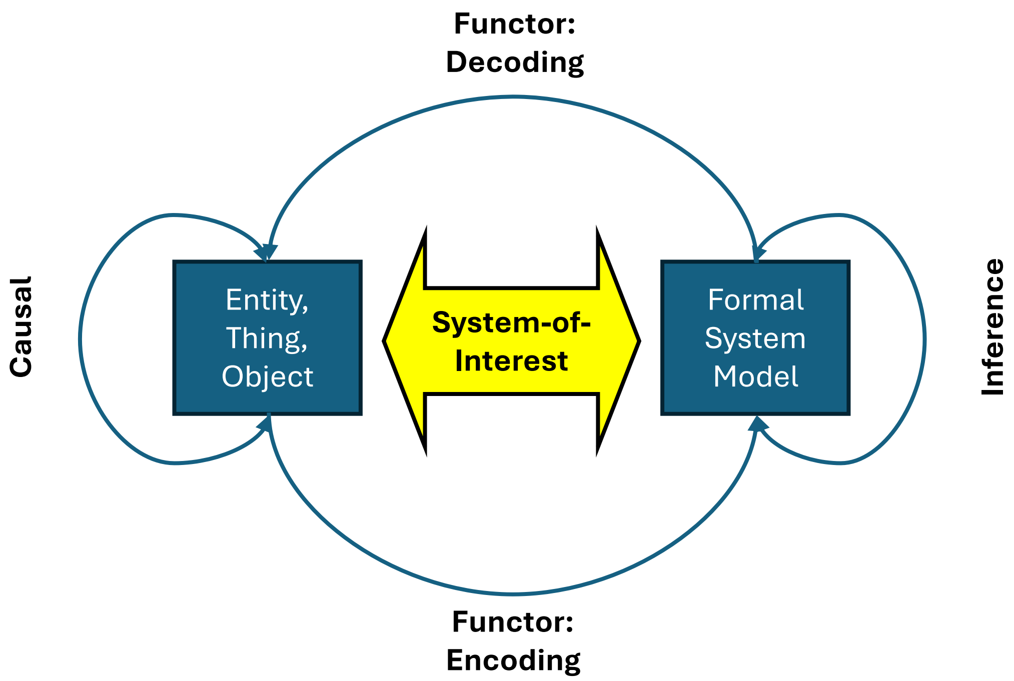
Tom Marzolf Model of a Molecule

H_2O can be modelled as a Molecule using the above model.

Rosen Model: Water Molecule as a System

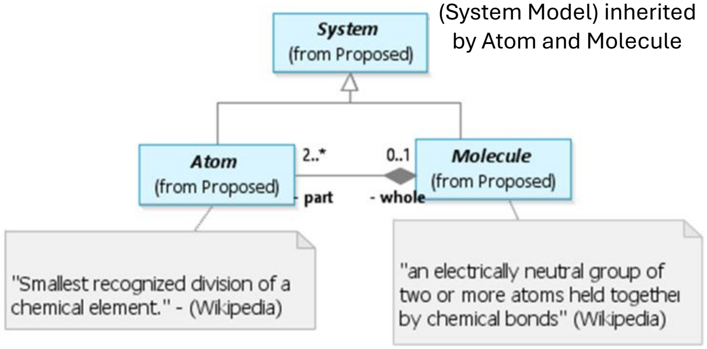
Entity, Thing, Object:
Water Molecule:
H_2O

The Entity, Thing, Object is H2O
As a system-of-interest.



A system "lens" to see an Entity, Thing or Object as a system.

Definition of System (System Model) inherited by Atom and Molecule



Tom Marzolf Model of a Molecule as a system

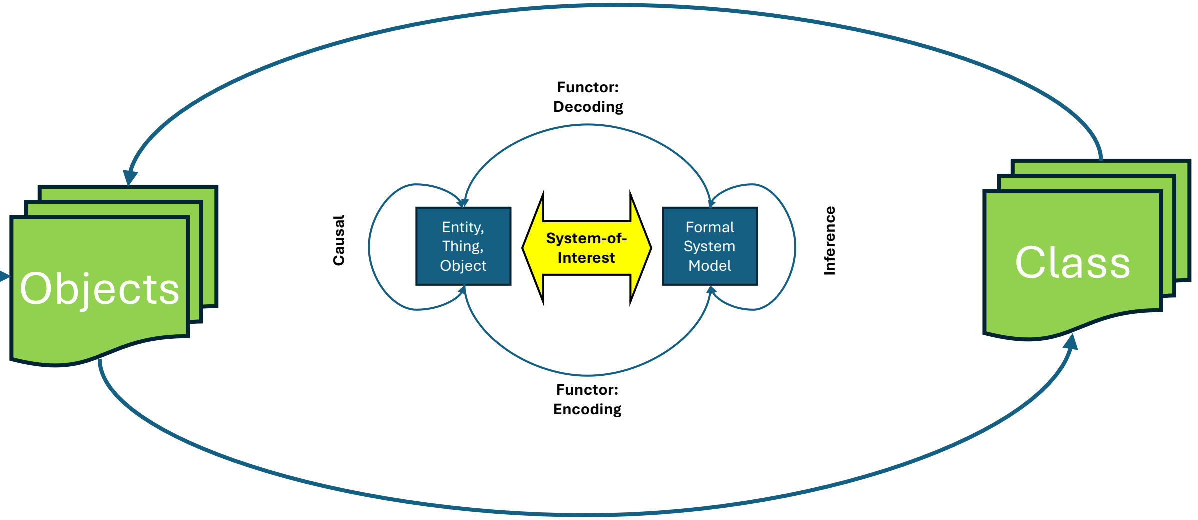
H_2O can be modelled using the Molecule Model as a System.

UML and Modeling Relation

Matlab, Simulink, open Modelica capabilities: Simulate Behavioural diagrams based in UML (activity, sequence, state machine)
Provide discrete, stochastic and continuous simulation.

Behavioral Models (Activity, Sequence, etc) connected with Matlab or Simulink provide a means to check / predict the behavior of the objects

Behaviorial Diagrams (Activity, state machines or Sequence Diagrams) are used to model performance.
These models can be connected with MATLAB or SIMULINK to carry out performance models.



Object Diagrams are Structural Models used for instantiation of a class model

Atom may be an Abstract Class while an Oxygen Atom is an Instance of the Atom Class with typical numbers of protons, neutrons, and electrons.

Classes formed through observation and measurement of objects to define Attributes and Functions

Class Diagrams are Structural Models

To identify attribute and function names (ontology)

Classes may be Abstract or Concrete

Abstract Classes require object re-definition to form an instance

Conclusions: So far .. More to come.

- **Entity, Thing, Object**
  - The focus for scientific understanding and learning is an Entity, Thing, Object.
  - An Entity, Thing or object has characteristics / properties that can be observed / named.
  - These characteristics / properties may not be described in system terms (e.g. may not have inherent named system concepts).
- **Models are used to understand an Entity, Thing, Object.**
  - Models reflect our understanding of an Entity, Thing, Object
  - Models can be improved (we now have a Subatomic way to understand Molecules and Atoms)
  - In most cases, models are not the Entity, Thing, Object. The Map is not the Territory.
  - Classifications of Entities, Things, Objects is another type of model.
- **System Models are used to understand an Entity, Thing, Object <u>as a System</u>**
  - Using a common language and representation
  - Definition of System and an associated Abstract System model are key.
  - Using a formal set of tools and techniques (math, modelling, etc).
  - Along with a system classification model.