DOMAIN SCIENCE & ENGINEERING

The TU Wien Lectures, Fall 2022

—

Dines Bjørner

Technical University of Denmark

The Triptych Dogma

In order to *specify* **software**,
we must understand its requirements.

In order to *prescribe* **requirements**
we must understand the **domain**.

So we must **study, analyse** and **describe** domains.

Day #1: Domains

- In this lecture, i.e., the next 45 mins.,
  - I shall survey "all" of the most important
  - aspect of Domain Analysis & Description.
- These will all be further explained
  and more aspects will be introduced
  in all the subsequent lectures.

CHAPTER 3. Domains

3.1 Domain Definition

Definition 1 . *Domain:* By a *domain* we shall understand

- a *rationally describable* segment of
- a *discrete dynamics* segment of
- a *human assisted reality*, i.e., of the world;
- its *more-or-less related solid or fluid entities*:
  - *natural* [“God-given”] and
  - *artefactual* [“man-made”],
- and its *living species entities*:
- *plants* and *animals* – including, notably, *humans* ■

Example 1 . Domains: A few, more-or-less self-explanatory examples:

- **Rivers –** with their natural sources, deltas, tributaries, waterfalls, etc., and their man-made dams, harbours, locks, etc. [19]
- **Road nets –** with street segments and intersections, traffic lights, and automobiles.
- **Pipelines –** with their wells, pipes, valves, pumps, forks, joins and wells [8].
- **Container terminals –** with their container vessels, containers, cranes, trucks, etc. [14] ■

- The definition relies on the understanding of the terms

  'rationally describable', *'discrete dynamics'*, *'human assisted'*, *'solid'* and *'fluid'*.

  The last two will be explained later.
- By ***rationally describable*** we mean that what is described can be understood, including reasoned about, in a rational, that is, logical manner.
- By ***discrete dynamics*** we imply that we shall basically rule out such domain phenomena which have properties which are continuous with respect to their time-wise behaviour.
- By ***human-assisted*** we mean that the domains – that we are interested in modelling – have, as an important property, that they possess man-made entities.

3.2 Phenomena and Entities

- **Definition 2 .** ***Phenomena:*** By a *phenomenon* we shall understand a fact that is observed to exist or happen ■
  - Some phenomena are rationally describable – to a large or full degree – others are not.
- **Definition 3 .** ***Entities:*** By an *entity* we shall understand a more-or-less rationally describable phenomenon ■
- **Example 2 . Phenomena and Entities:** Some, but not necessarily all aspects of a river can be rationally described, hence can be still be considered entities. Similarly, many aspects of a road net can be rationally described, hence will be considered entities ■

3.3 Endurants and Perdurants

3.3.1 Endurants

- **Definition 4** . ***Endurants:*** those quantities of domains that we can observe (see and touch), in *space*, as "complete" entities at no matter which point in *time* – "material" entities that persists, endures ■

  Example 3 . **Endurants:** a street segment [link], a street intersection [hub], an automobile ■

- Domain endurants, when eventually modelled in software, typically become data. Hence the careful analysis of domain endurants is a prerequisite for subsequent careful conception and analyses of data structures for software, including data bases.

3.3.2 Perdurants

- **Definition 5 .** *Perdurants* those quantities of domains for which only a fragment exists, in *space*, if we look at or touch them at any given snapshot in *time* ■

  Example 4 . Perdurant: a moving automobile ■

- Domain perdurants, when eventually modelled in software, typically become processes. Hence the careful analysis of domain perdurants is a prerequisite for subsequent careful conception and analyses of functions (procedures).

3.4 External and Internal Endurant Qualities

3.4.1 External Qualities

Definition 6 . ***External qualities:*** of endurants of a manifest domain

- are, in a simplifying sense, those we can
  - sea,
  - touch and
  - have spatial extent.
- They, so to speak, take form.

Example 5 . External Qualities:

- The Cartesian[1]
  - of sets of solid atomic street intersections, and
  - of sets of solid atomic street segments, and
  - of sets of solid automobiles

  of a road transport system
- where the
  - Cartesian,
  - sets,
  - atomic, and
  - solid

  reflect external qualities ■

[1] Cartesian after the French philosopher, mathematician, scientist René de Descartes (1596–1650)

3.4.1.1 Discrete or Solid Endurants

Definition 7 . ***Discrete or Solid Endurants:*** By a *solid* [or *discrete*] endurant we shall understand an endurant

- which is separate, individual or distinct in form or concept,
- or, rephrasing: have 'body' [or magnitude] of three-dimensions: length, breadth and depth [34, Vol. II, pg. 2046] ■

Example 6 . Solid Endurants:

- The
  - wells,
  - pipes,
  - valves,
  - pumps,
  - forks,
  - joins and
  - sinks

  of pipelines are solids.
- [These units may, however, and usually will, contain fluids, e.g., oil, gas or water] ■

- We shall mostly be analysing and describing solid endurants.
- As we shall see, in the next section,
  - we analyse and describe solid endurants as
  - either parts
  - or living species: animals and humans.
- We shall mostly be concerned with parts.
  - That is, we shall just, as: “in passing”,
  - for sake of completeness,
  - mention living species !

3.4.1.2 Fluids

- **Definition 8 .** *Fluid Endurants:*
  - By a *fluid endurant* we shall understand an endurant which is
    - ⋆ prolonged, without interruption, in an unbroken series or pattern;
    - ⋆ or, rephrasing: a substance (liquid, gas or plasma) having the property of flowing, consisting of particles that move among themselves [34, Vol. I, pg. 774] ■

Example 7 . Fluid Endurants:

- water,
- oil,
- gas,
- compressed air,
- smoke ■

- Fluids are otherwise
  - liquid, or
  - gaseous, or
  - plasmatic, or
  - granular[2], or
  - plant products, i.e., chopped sugar cane, threshed, or otherwise[3],
  - et cetera.
- Fluid endurants will be analysed and described in relation to solid endurants, viz. their "containers".

[2] This is a purely pragmatic decision. "Of course" sand, gravel, soil, etc., are not fluids, but for our modelling purposes it is convenient to "compartmentalise" them as fluids !
[3] See footnote 2.

3.4.1.3 Parts

- **Definition 9 .** ***Parts:***
  - The non-living species solids are what we shall call parts ■
- Parts are the "work-horses" of man-made domains.
- That is, we shall mostly be concerned
  with the analysis and description of endurants into parts.

Example 8 . Parts: The previous example of solids was also an example of parts ■

- We distinguish between atomic and compound parts.

3.4.1.3.1 Atomic Parts

Definition 10 . *Atomic Part, I:*

- By an *atomic part* we shall understand a part
  - which the domain analyser considers to be indivisible
  - in the sense of not meaningfully,
  - for the purposes of the domain under consideration,
  - that is, to not meaningfully consist of sub-parts ■

3.4.1.3.2 Compound Parts

- We, pragmatically, distinguish between
  - Cartesian-product-, and
  - set-

  oriented parts.
- If Cartesian-oriented, to consist of two or more distinctly sort-named endurants (solids or fluids),
- If set-oriented, to consist of an indefinite number of zero, one or more parts.

Definition 11 . *Compound Part, I:*

- *Compound part*s are those which are
  - either Cartesian-product-
  - or are set-
- oriented parts ■

Example 9 . Compound Parts: A road net consisting of

- a set of hubs, i.e., street intersections or “end-of-streets”, and
- a set of links, i.e., street segments (with no contained hubs),

is a Cartesian compound;

- and the sets of hubs and the sets of links

are part set compounds ■

3.4.2 An Aside: An Upper Ontology

- We have been reasonably careful
  - to just introduce and state informal definitions
  - of phenomena and some classes thereof.
- In the next chapter we shall, in a sense, "repeat" coverage of these phenomena.
  - But now in a more analytic manner.
  - Figure 3.1 on the next slide is intended to indicate this.

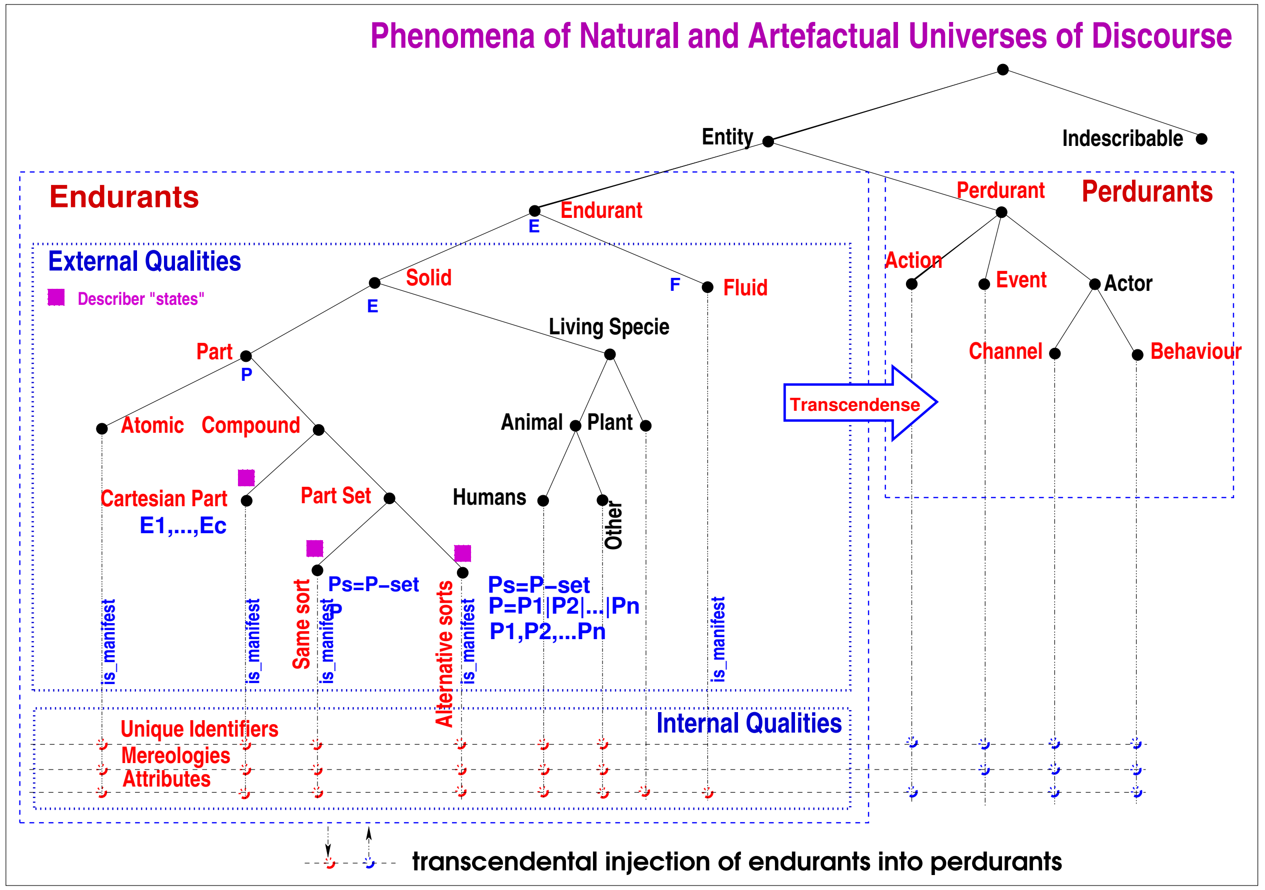

Figure 3.1: Upper Ontology

- So far we have only touched upon
  - the ‘External Qualities’ labeled, dotted-dashed box
  - of the ‘Endurants’ label-led dashed box of Fig. 3.1.
- In Chapter 4 we shall treat external qualities in more depth —
- more systematically: analytically and descriptionally.

3.4.3 Internal quality

Definition 12 . *Internal qualities:*

- those properties [of endurants]
- that do not occupy *space*
- but can be measured or spoken about ■

Example 10 . Internal qualities:

- the unique identity of a part,
- the relation of part to other parts, and
- the endurant attributes such as temperature, length, colour ■

3.4.3.1 Unique identity

- **Definition 13 .** *Unique identity:*
  - an immaterial property
  - that distinguishes two *spatially* distinct solids ■

Example 11 . Unique identities:

- Each hub in a road net is unique identified,
- so is each link
- and automobile ■

3.4.3.2 Mereology

- **Definition 14 .** ***Mereology:*** a theory of [endurant] part-hood relations:
  - of the relations of an [endurant] parts to a whole
  - and the relations of [endurant] parts to [endurant] parts within that whole ■

Example 12 . Mereology:

- that a link is topologically *connected* to exactly two specific hubs,
- that hubs are *connected* to zero, one or more specific links,
- and that links and hubs are *open* to specific subsets of automobiles ■

3.4.3.3 Attribute

Definition 15 . ***Attributes:*** Properties of endurants

- that are not *spatially* observable,
- but can be either physically
- (electronically, chemically, or otherwise)
- measured or can be objectively spoken about ■

Example 13 . Attribute: Links have

- lengths, and,
- at any one time,
- zero, one or more automobiles are occupying the links ■

3.5 Prompts

3.5.1 Analysis Prompts

- **Definition 16 .** *Analysis prompt:*
  - a predicate or a function
  - that may be posed by humans to facets of a domain.
  - Observing the domain the analyser may then
  - act upon the combination of the particular prompt
  - (whether a predicate or a function,
    and then what particular one of these it is)
  - thus “applying” it to a domain phenomena,
  - and yielding, in the minds of the humans,
  - either a truth value or some other form of value ■

3.5.1.1 Analysis Predicate

- **Definition 17 .** *Analysis predicates:*
  - an analysis prompt
  - which yields a truth value ■

Example 14 . Analysis Predicates: General examples are

- can an observable phenomena be rationally described, i.e., an entity,
- is an entity a solid or a fluid.
- is a solid endurant a part or a living species ■

3.5.1.2 Analysis Function

Definition 18 . *Analysis function:*

- an analysis prompt which yields some RSL-**Text** ■

Example 15 . Analysis Functions: Two examples:

- one yields the endurants of a Cartesian part and their respective sort names,
- another yields the set of a parts of a part set and their common type ■

3.5.2 Description Prompt

- **Definition 19** . *Description prompt:*
  - a function that may be posed by humans
  - who may then act upon it:
    - ⋆ "applying" it to a domain phenomena, and
    - ⋆ "yielding" narrative and formal RSL-**Text**s describing what is being observed ■

Example 16 . Description Prompts:

- result in RSL-**Text**s describing for example a
  - (i) Cartesian endurant, or
  - (ii) its unique identifier,
  - (iii) or its mereology, or
  - (iv) its attributes,
  - (iv) or other ■

3.6 Perdurant Concepts

3.6.1 “Morphing” Parts into Behaviours

- As already indicated we shall
  - transcendentally deduce
  - (perdurant) behaviours from
  - those (endurant) parts
    - ⋆ which we, as domain analysers cum describers,
    - ⋆ have endowed with all three kinds of internal qualities:
    - ⋆ unique identifiers, mereologies and attributes.
- Chapter 6, will show how.

3.6.2 State

Definition 20 . *State:*

- A state is any set of the parts of a domain ■

Example 17 . A Road System State: The domain analyser cum describer may,
In brief, decide that a road system state consists of

- the road net aggregate (of hubs and links)[4],
- all the hubs, and all the links, and
- the automobile aggregate (of all the automobiles)[5], and
- all the individual automobiles ■

[4] The road net aggregate, in its perdurant form, may "model" the *Department of Roads* of some country, province, or town.
[5] The automobile aggregate aggregate, in its perdurant form, may "model" the *Department of Vehicles* of some country, province, or town.

3.6.3 Actors

Definition 21 . ***Actors:***

- An actor is anything that can initiate an action, an event or a behaviour ■

3.6.3.1 Action

Definition 22 . ***Actions:***

- An action is a function that can purposefully change a state ■

Example 18 . Road Net Actions: These are some road net actions:

- The insertion of a new or removal of an existing hub; or
- the insertion of a new, or removal of an existing link;

3.6.3.2 Event

Definition 23 . ***Events:***

- An event is a function that surreptitiously changes a state ■

Example 19 . Road Net Events: These are some road net events:

- The blocking of a link due to a mud slide;
- the failing of a hub traffic signal due to power outage;
- the blocking of a link due to an automobile accident.

3.6.3.3 Behaviour

Definition 24 . ***Behaviours***

- a behaviour is a set of sequences of actions, events and behaviours
  ■

Example 20 . Road Net Traffic:

- Road net traffic can be seen as a behaviour
  - of all the behaviours of automobiles,
    - ⋆ where each automobile behaviour is seen as sequence of start, stop, turn right, turn left, etc., actions;

- of all the behaviours of links
  - where each link behaviour is seen
    as a set of sequences (i.e., behaviours) of “following” the
    - link entering,
    - link leaving, and
    - movement

    of automobiles on the link;
- of all the behaviours of hubs (etc.);
- of the behaviour of the aggregate of roads,
  viz. *The Department of Roads*, and
- of the behaviour of the aggregate of automobiles,
  viz, *The Department of Vehicles*.

3.6.4 Channel

- **Definition 25 .** *Channel:*
  - A channel is anything
  - that allows synchronisation and communication
  - of values
  - between two behaviours ■

- We shall use Tony Hoare's CSP concept [32]
  - to express synchronisation and communication of values between behaviours *i* and *j*.
  - Hence the behaviour *i* statement
    ch[j] ! value
    to state that behaviour *i*
    offers, "outputs": !, *value*
    to behaviours indicated by *j*.
  - And behaviour i expresses
    ch[j] ?
    that it is willing to accept
    "input from & synchronise with" behaviour *i*, ?,
    any *value*.

3.7 Domain Analysis & Description

3.7.1 Domain Analysis

Definition 26 . *Domain Analysis*

- is the act of studying a domain
- as well as the result of that study
- in the form of informal statements ■

3.7.2 Domain Description

Definition 27 . *Domain Description*

- is the act of describing a domain
- as well as the result of that act
- in the form of narratives and formal RSL-Text ■

3.8 Closing

- This lecture has introduced
  the main concepts of domains such as we shall treat
  (analyse and describe) domains.[6]
- The next lectures shall now systematically treat
  the analysis and description of domains.
  - That treatment takes concept by concept and
    - ⋆ provides proper definitions and
    - ⋆ introduces appropriate analysis and description prompts;
    - ⋆ one-by-one, in an almost pedantic,
    - ⋆ hence perhaps "slow" progression !

[6] We have omitted treatment of *living species: plants* and *animals* – the latter including *humans*. They will be treated in the next chapter !

- The student may be excused
  - if they, now-and-then, loose sight of "their way".
- Hence the present chapter.
  - To show "the way":
  - that, for example,
    when we treat external endurant qualities,
  - there is still the internal endurant qualities,
  - and that the whole thing leads of to perdurants:
    - ⋆ actors,
    - ⋆ actions,
    - ⋆ events and
    - ⋆ behaviours.

THANKS

- The next 45 minute lecture shall present fragments of a *road transport system* example.

Appendix A. Road Transport

A.1 The Road Transport Domain

- Our universe of discourse in this chapter is the road transport domain.
- Not a specific one, but "a generic road transport domain".

A.1.1 Naming

type RTS

A.1.2 Rough Sketch

- The generic road transport domain that we have in mind consists of
  - a road net (aggregate) and
  - an aggregate of vehicles
  - such that the road net serves to convey vehicles.
- We consider the road net to consist of
  - hubs, i.e., street intersections,
    or just street segment connection points, and
  - links, i.e., street segments between adjacent hubs.

- We consider the aggregate of vehicles to include
  - in addition to vehicales, i.e., automobiles,
  - a department of motor vehicles (DMVs),
  - zero or more bus companies, each with zero, one or more buses, and
  - vehicle associations, each with
    - ⋆ zero, one or more members
    - ⋆ who are owners of zero, one or more vehicles[1] ■

[1] This “rough” narrative fails to narrate what ...

A.2 External Qualities

A Road Transport System, I – Manifest External Qualities:

- Our intention is that the manifest external qualities of a road transport system are those of its
  - roads,
    - their **hubs** i.e., road (or street) intersections, and
    - their **links**, i.e., the roads (streets) between hubs, and
  - **vehicles**, i.e., automobiles – that ply the roads –
    - the buses, trucks, private cars, bicycles, etc. ■

A.2.1 A Road Transport System, II – Abstract External Qualities

- Examples of what could be considered abstract external qualities of a road transport domain are:
  - the aggregate of all hubs and all links,
  - the aggregate of all buses, say into bus companies,
  - the aggregate of all bus companies into public transport, and
  - the aggregate of all vehicles into a department of vehicles.
- Some of these aggregates may, at first be treated as abstract.
- Subsequently, in our further analysis & description we may decide to consider some of them as concretely manifested in, for example, actual
  - departments of roads.

A.2.2 Transport System Structure

- A transport system is modeled as structured into
  - a *road net structure* and
  - an *automobile structure*.
- The *road net structure* is then structured as a pair:
  - a *structure of hubs* and
  - a *structure of links*.
- These latter structures are then modeled as set of hubs, respectively links.

- We could have modeled the road net *structure*
  - as a *composite part*
  - with *unique identity, mereology* and *attributes*
  - which could then serve to model
  - a `road net authority`.
- And we could have modeled the automobile *structure*
  - as a *composite part*
  - with *unique identity, mereology* and *attributes*
  - which could then serve to model
  - a `department of vehicles` ■

A.2.3 Atomic Road Transport Parts

- From one point of view all of the following can be considered atomic parts:
  - hubs,
  - links[2], and
  - automobiles.

[2] Hub ≡ street intersection; link ≡ street segments with no intervening hubs.

A.2.4 Compound Road Transport Parts

A.2.4.1 The Composites

182. There is the *universe of discourse*, UoD.

It is structured into

183. a *road net*, RN, and
184. a *fleet of vehicles*, FV.

Both are structures. ....

type
182 UoD **axiom** $\forall$ uod:UoD · is_structure(uod).
183 RN **axiom** $\forall$ rn:RN · is_structure(rn).
184 FV **axiom** $\forall$ fv:FV · is_structure(fv).
value
183 obs_RN: UoD $\rightarrow$ RN
184 obs_FV: UoD $\rightarrow$ FV ■

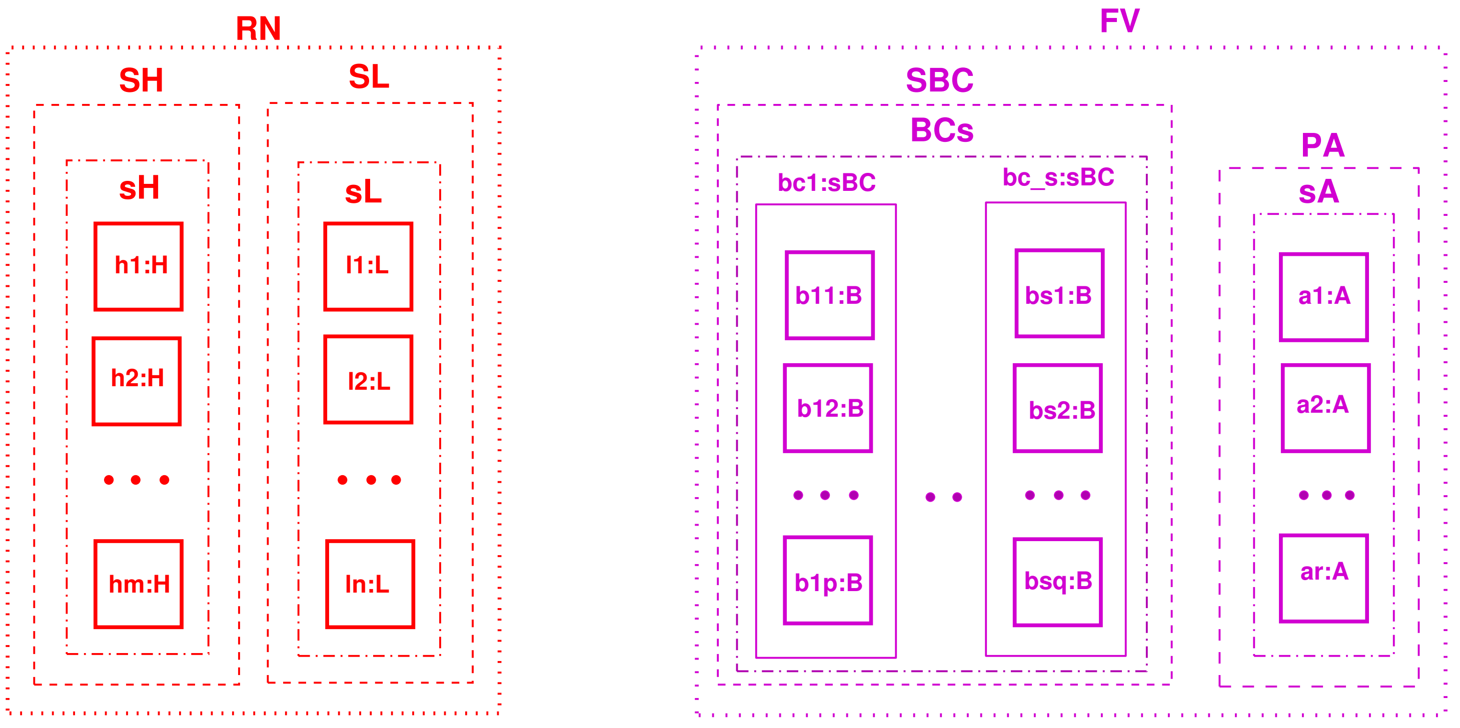

Figure A.1: A Road Transport System Compounds and Structures

A.2.4.2 The Part Parts

185. The structure of hubs is a set, sH, of atomic hubs, H.
186. The structure of links is a set, sL, of atomic links, L.
187. The structure of buses is a set, sBC, of composite bus companies, BC.
188. The composite bus companies, BC, are sets of buses, sB.
189. The structure of private automobiles is a set, sA, of atomic automobiles, A.

type

185 H, sH = H-**set** **axiom** ∀ h:H · is_atomic(h)
186 L, sL = L-**set** **axiom** ∀ l:L · is_atomic(l)
187 BC, BCs = BC-**set** **axiom** ∀ bc:BC · is_composite(bc)
188 B, Bs = B-**set** **axiom** ∀ b:B · is_atomic(b)
189 A, sA = A-**set** **axiom** ∀ a:A · is_atomic(a)

value

185 obs_sH: SH → sH
186 obs_sL: SL → sL
187 obs_sBC: SBC → BCs
188 obs_Bs: BCs → Bs
189 obs_sA: SA → sA ■

A.2.5 The Transport System State

190. Let there be given a universe of discourse, *rts*. It is an example of a state.

From that state we can calculate other states.

191. The set of all hubs, *hs*.
192. The set of all links, *ls*.
193. The set of all hubs and links, *hls*.
194. The set of all bus companies, *bcs*.
195. The set of all buses, *bs*.
196. The set of all private automobiles, *as*.
197. The set of all parts, *ps*.

value

190 rts:UoD [34]

191 hs:H-**set** ≡:H-**set** ≡ obs_sH(obs_SH(obs_RN(rts)))

192 ls:L-**set** ≡:L-**set** ≡ obs_sL(obs_SL(obs_RN(rts)))

193 hls:(H|L)-**set** ≡ $hs \cup ls$

194 bcs:BC-**set** ≡ obs_BCs(obs_SBC(obs_FV(obs_RN(rts))))

195 bs:B-**set** ≡ ∪{obs_Bs(bc)|bc:BC•bc ∈ bcs}

196 as:A-**set** ≡ obs_BCs(obs_SBC(obs_FV(obs_RN(rts))))

197 ps:(UoB|H|L|BC|B|A)-**set** ≡ $rts \cup hls \cup bcs \cup bs \cup as$

A.3 Internal Qualities

A.3.1 Unique Identifiers

198. We assign unique identifiers to all parts.
199. By a road identifier we shall mean a link or a hub identifier.
200. By a vehicle identifier we shall mean a bus or an automobile identifier.
201. Unique identifiers uniquely identify all parts.
    (a) All hubs have distinct [unique] identifiers.
    (b) All links have distinct identifiers.
    (c) All bus companies have distinct identifiers.
    (d) All buses of all bus companies have distinct identifiers.
    (e) All automobiles have distinct identifiers.
    (f) All parts have distinct identifiers.

type

198 H_UI, L_UI, BC_UI, B_UI, A_UI

199 R_UI = H_UI | L_UI

200 V_UI = B_UI | A_UI

value

201a uid_H: H → H_UI

201b uid_L: H → L_UI

201c uid_BC: H → BC_UI

201d uid_B: H → B_UI

201e uid_A: H → A_UI

A.3.1.1 Extract Parts from Their Unique Identifiers

202. From the unique identifier of a part we can retrieve, $\wp$, the part having that identifier.

type
202 P = H | L | BC | B | A
value
202 $\wp$: H_UI→H | L_UI→L | BC_UI→BC | B_UI→B | A_UI→A
202 $\wp$(ui) ≡ **let** p:(H|L|BC|B|A)·p∈ps∧uid_P(p)=ui **in** p **end**

A.3.1.2 All Unique Identifiers of a Domain

We can calculate:

203. the *s*et, $h_{ui}s$, of *u*nique *h*ub *i*dentifiers;
204. the *s*et, $l_{ui}s$, of *u*nique *l*ink *i*dentifiers;
205. the *m*ap, $hl_{ui}m$, from *u*nique *h*ub *i*dentifiers to the *s*et of *u*nique *l*ink *i*identifiers of the links connected to the zero, one or more identified hubs,
206. the *m*ap, $lh_{ui}m$, from *u*nique *l*ink *i*dentifiers to the *s*et of *u*nique *h*ub *i*identifiers of the two hubs connected to the identified link;
207. the *s*et, $r_{ui}s$, of all *u*nique hub and link, i.e., *r*oad *i*dentifiers;
208. the *s*et, $bc_{ui}s$, of *u*nique *b*us *c*ompany *i*dentifiers;

209. the *s*et, $b_{ui}s$, of *u*nique *b*us *i*dentifiers;
210. the *s*et, $a_{ui}s$, of *u*nique private *a*utomobile *i*dentifiers;
211. the *s*et, $v_{ui}s$, of *u*nique bus and automobile, i.e., *v*ehicle *i*dentifiers;
212. the *m*ap, $bcb_{ui}m$, from *u*nique *b*us *c*ompany *i*dentifiers to the *s*et of its *u*nique *b*us *i*dentifiers; and
213. the (*b*ijective) *m*ap, $bbc_{ui}bm$, from *u*nique *b*us *i*dentifiers to their *u*nique *b*us *c*ompany *i*dentifiers.

value

203 $h_{ui}s$:H_UI-**set** ≡ {uid_H(h)|h:H·h ∈ hs}
204 $l_{ui}s$:L_UI-**set** ≡ {uid_L(l)|l:L·l ∈ ls}
207 $r_{ui}s$:R_UI-**set** ≡ $h_{ui}s \cup l_{ui}s$
205 $hl_{ui}m$:(H_UI $\overrightarrow{m}$ L_UI-**set**) ≡
205 [h_ui↦luis|h_ui:H_UI,luis:L_UI-**set**·h_ui∈$h_{ui}s$∧(_,luis,_)=mereo_H(η(h_ui))] [cf. Ite
206 $lh_{ui}m$:(L+UI $\overrightarrow{m}$ H_UI-**set**) ≡
206 [l_ui↦huis | h_ui:L_UI,huis:H_UI-**set** · l_ui∈$l_{ui}s$ ∧ (_,huis,_)=mereo_L(η(l_ui))] [cf.
208 $bc_{ui}s$:BC_UI-**set** ≡ {uid_BC(bc)|bc:BC·bc ∈ bcs}
209 $b_{ui}s$:B_UI-**set** ≡ ∪{uid_B(b)|b:B·b ∈ bs}
210 $a_{ui}s$:A_UI-**set** ≡ {uid_A(a)|a:A·a ∈ as}
211 $v_{ui}s$:V_UI-**set** ≡ $b_{ui}s \cup a_{ui}s$
212 $bcb_{ui}m$:(BC_UI $\overrightarrow{m}$ B_UI-**set**) ≡
212 [bc_ui ↦ buis | bc_ui:BC_UI, bc:BC · bc∈bcs ∧ bc_ui=uid_BC(bc) ∧ (_,_,buis)=mer
213 $bbc_{ui}bm$:(B_UI $\overrightarrow{m}$ BC_UI) ≡
213 [b_ui ↦ bc_ui | b_ui:B_UI,bc_ui:BC_ui · bc_ui=**dom** $bcb_{ui}m$∧b_ui∈$bcb_{ui}m$(bc_ui)]

A.3.1.3 Uniqueness of Road Net Identifiers

- We must express the following axioms:

214. All hub identifiers are distinct.
215. All link identifiers are distinct.
216. All bus company identifiers are distinct.
217. All bus identifiers are distinct.
218. All private automobile identifiers are distinct.
219. All part identifiers are distinct.

axiom

214 $\textbf{card}\ hs = \textbf{card}\ h_{ui}s$

215 $\textbf{card}\ ls = \textbf{card}\ l_{ui}s$

216 $\textbf{card}\ bcs = \textbf{card}\ bc_{ui}s$

217 $\textbf{card}\ bs = \textbf{card}\ b_{ui}s$

218 $\textbf{card}\ as = \textbf{card}\ a_{ui}s$

219 $\textbf{card}\ \{h_{ui}s \cup l_{ui}s \cup bc_{ui}s \cup b_{ui}s \cup a_{ui}s\}$

219 $\quad = \textbf{card}\ h_{ui}s + \textbf{card}\ l_{ui}s + \textbf{card}\ bc_{ui}s + \textbf{card}\ b_{ui}s + \textbf{card}\ a_{ui}s$ ■

A.3.2 Mereology

A.3.2.1 Mereology Types and Observers

220. The mereology of hubs is a pair: (i) the set of all bus and automobile identifiers[3], and (ii) the set of unique identifiers of the links that it is connected to and the set of all unique identifiers of all vehicles (buses and private automobiles).[4]
221. The mereology of links is a pair: (i) the set of all bus and automobile identifiers, and (ii) the set of the two distinct hubs they are connected to.
222. The mereology of a bus company is a set the unique identifiers of the buses operated by that company.
223. The mereology of a bus is a pair: (i) the set of the one single unique identifier of the bus company it is operating for, and (ii) the unique identifiers of all links and hubs[5].
224. The mereology of an automobile is the set of the unique identifiers of all links and hubs[6].

type
220 H_Mer = V_UI-**set**×L_UI-**set**
221 L_Mer = V_UI-**set**×H_UI-**set**
222 BC_Mer = B_UI-**set**
223 B_Mer = BC_UI×R_UI-**set**
224 A_Mer = R_UI-**set**

value
220 mereo_H: H → H_Mer
221 mereo_L: L → L_Mer
222 mereo_BC: BC → BC_Mer
223 mereo_B: B → B_Mer
224 mereo_A: A → A_Mer

A.3.2.2 Invariance of Mereologies

- For mereologies one can usually express some invariants.
  - Such invariants express *"law-like properties"*,
  - facts which are indisputable.

[3] This is just another way of saying that the meaning of hub mereologies involves the unique identifiers of all the vehicles that might pass through the hub `is_of_interest` to it.
[4] The link identifiers designate the links, zero, one or more, that a hub is connected to `is_of_interest` to both the hub and that these links is `interested` in the hub.
[5] — that the bus might pass through
[6] — that the automobile might pass through

A.3.2.2.1 Invariance of Road Nets

- The observed mereologies must express identifiers of the state of such for road nets:

axiom

220 ∀ (vuis,luis):H_Mer · luis⊆$l_{ui}s$ ∧ vuis=$v_{ui}s$
221 ∀ (vuis,huis):L_Mer · vuis=$v_{ui}s$ ∧ huis⊆$h_{ui}s$ ∧ **card**huis=2
222 ∀ buis:H_Mer · buis = $b_{ui}s$
223 ∀ (bc_ui,ruis):H_Mer·bc_ui∈$bc_{ui}s$∧ruis=$r_{ui}s$
224 ∀ ruis:A_Mer · ruis=$r_{ui}s$

225. For all hubs, h, and links, l, in the same road net,
226. if the hub h connects to link l then link l connects to hub h.

axiom
225 ∀ h:H,l:L · h ∈ hs ∧ l ∈ ls ⇒
225   **let** (_,luis)=mereo_H(h), (_,huis)=mereo_L(l)
226   **in** uid_L(l)∈luis ≡ uid_H(h)∈huis **end**

227. For all links, l, and hubs, h_a, h_b, in the same road net,
228. if the l connects to hubs h_a and h_b, then h_a and h_b both connects to link l.

axiom

227 ∀ h_a,h_b:H,l:L · {h_a,h_b} ⊆ *hs* ∧ l ∈ *ls* ⇒
227 **let** (_,luis)=mereo_H(h), (_,huis)=mereo_L(l)
228 **in** uid_L(l)∈luis ≡ uid_H(h)∈huis **end**

A.3.2.2.2 Possible Consequences of a Road Net Mereology

229. are there [isolated] units from which one can not "reach" other units ?
230. does the net consist of two or more "disjoint" nets ?
231. et cetera.

- We leave it to the reader to narrate and formalise the above properly.

A.3.2.2.3 Fixed and Varying Mereology

- Let us consider a road net.
  - If hubs and links never change "affiliation", that is:
    - ⋆ hubs are in fixed relation to zero one or more links, and
    - ⋆ links are in a fixed relation to exactly two hubs
    - ⋆ then the mereology is a *fixed mereology*.

- If, on the other hand
  - ⋆ hubs may be inserted into or removed from the net, and/or
  - ⋆ links may be removed from or inserted between any two existing hubs,
  - ⋆ then the mereology is a *varying mereology*.

A.3.3 Attributes

A.3.3.1 Hub Attributes

- We treat some attributes of the hubs of a road net.

232. There is a hub state.
    - It is a set of pairs, (l_f, l_t), of link identifiers,
        - where these link identifiers are in the mereology of the hub.
    - The meaning of the hub state
        - in which, e.g., (l_f, l_t) is an element,
        - is that the hub is open, **"green"**,
        - for traffic *f*rom link l_f *t*o link l_t.
        - If a hub state is empty
        - then the hub is closed, i.e., **"red"**
        - for traffic from any connected links to any other connected links.

233. There is a hub state space.
   - It is a set of hub states.
   - The current hub state must be in its state space.
   - The meaning of the hub state space is
     – that its states are all those the hub can attain.
234. Since we can think rationally about it,
   - it can be described, hence we can model, as an attribute of hubs, a history of its traffic:
     – the recording, per unique bus and automobile identifier,
     – of the time ordered presence in the hub of these vehicles.
   - Hub history is an *event history*.

type
232 $H\Sigma$ = (L_UI×L_UI)-**set**
axiom
232 ∀ h:H • obs_$H\Sigma$(h) ∈ obs_$H\Omega$(h)
type
233 $H\Omega$ = $H\Sigma$-**set**
234 H_Traffic
234 H_Traffic = (A_UI|B_UI) $\overrightarrow{m}$ ($\mathbb{TIME}$ × VPos)*
axiom
234 ∀ ht:H_Traffic,ui:(A_UI|B_UI) •
234   ui ∈ **dom** ht ⇒ time_ordered(ht(ui))
value
232 attr_$H\Sigma$: H → $H\Sigma$
233 attr_$H\Omega$: H → $H\Omega$
234 attr_H_Traffic: H → H_Traffic
value
234 time_ordered: ($\mathbb{TIME}$ × VPos)* → **Bool**
234 time_ordered(tvpl) ≡ ...

- In Item 234 on the facing slide we model the time-ordered sequence of traffic as a discrete sampling, i.e., $\overrightarrow{m}$, rather than as a continuous function, →.

A.3.3.2 Invariance of Traffic States

235. The link identifiers of hub states must be in the set, $l_{ui}s$, of the road net's link identifiers.

axiom

235 $\forall$ h:H · h $\in$ hs $\Rightarrow$

235 **let** hσ = attr_HΣ(h) **in**

235 $\forall$ ($l_{ui}i$,$li_{ui}i'$):(L_UI×L_UI) · ($l_{ui}i$,$l_{ui}i'$) $\in$ hσ $\Rightarrow$ $\{l_{ui_i}, l'_{ui_i}\} \subseteq l_{ui}s$ **end**

A.3.3.3 Link Attributes

We show just a few attributes.

236. There is a link state. It is a set of pairs, (h_f,h_t), of distinct hub identifiers, where these hub identifiers are in the mereology of the link. The meaning of a link state in which (h_f,h_t) is an element is that the link is open, "**green**", for traffic *f* rom hub h_f *t*o hub h_t. Link states can have either 0, 1 or 2 elements.
237. There is a link state space. It is a set of link states. The meaning of the link state space is that its states are all those the which the link can attain. The current link state must be in its state space. If a link state space is empty then the link is (permanently) closed. If it has one element then it is a one-way link. If a one-way link, *l*, is imminent on a hub whose mereology designates that link, then the link is a "trap", i.e., a "blind cul-de-sac".

238. Since we can think rationally about it, it can be described, hence it can model, as an attribute of links a history of its traffic: the recording, per unique bus and automobile identifier, of the time ordered positions along the link (from one hub to the next) of these vehicles.
239. The hub identifiers of link states must be in the set, $h_{ui}s$, of the road net's hub identifiers.

type
236 LΣ = H_UI-**set**
axiom
236 ∀ lσ:LΣ·**card** lσ=2
236 ∀ l:L · obs_LΣ(l) ∈ obs_LΩ(l)
type
237 LΩ = LΣ-**set**
238 L_Traffic
238 L_Traffic = (A_UI|B_UI) $\xrightarrow[m]{}$ ($\mathbb{T}$×(H_UI×Frac×H_UI))*
238 Frac = **Real**, **axiom** frac:Fract · 0<frac<1
value
236 attr_LΣ: L → LΣ
237 attr_LΩ: L → LΩ
238 attr_L_Traffic: : → L_Traffic
axiom
238 ∀ lt:L_Traffic,ui:(A_UI|B_UI)·ui ∈ **dom** ht ⇒ time_ordered(ht(ui))
239 ∀ l:L · l ∈ ls ⇒
239 **let** lσ = attr_LΣ(l) **in** ∀ (h$_{ui}i$,h$_{ui}i'$):(H_UI×K_UI) ·
239 (h$_{ui}i$,h$_{ui}i'$) ∈ lσ ⇒ {h$_{ui_i}$,h$'_{ui_i}$} ⊆ h$_{ui}s$ **end**

A.3.3.4 Bus Company Attributes

- Bus companies operate a number of lines that service passenger transport along routes of the road net. Each line being serviced by a number of buses.

240. Bus companies create, maintain, revise and distribute
[to the public (not modeled here), and to buses]
bus time tables, not further defined.

type
240 BusTimTbl
value
240 attr_BusTimTbl: BC $\rightarrow$ BusTimTbl

- There are two notions of time at play here:
  - the indefinite "real" or "actual" time; and
  - the definite calendar, hour, minute and second time designation occurring in some textual form in, e.g., time tables.

A.3.3.5 Bus Attributes

We show just a few attributes.

241. Buses run routes, according to their line number, ln:LN, in the
242. bus time table, btt:BusTimTbl obtained from their bus company, and and keep, as inert attributes, their segment of that time table.
243. Buses occupy positions on the road net:
   (a) either *at a hub* identified by some h_ui,
   (b) or *on a link*, some *fraction*, f:Fract, down an *identified link, l_ui*, from one of its *identified connecting hubs*, fh_ui, in the direction of the other *identified hub*, th_ui.
244. Et cetera.

type
241 LN
242 BusTimTbl
243 BPos == atHub | onLink
243a atHub :: h_ui:H_UI
243b onLink :: fh_ui:H_UI×l_ui:L_UI×frac:Fract×th_ui:H_UI
243b Fract = **Real**, **axiom** frac:Fract · 0<frac<1
244 ...
value
242 attr_BusTimTbl: B → BusTimTbl
243 attr_BPos: B → BPos

A.3.3.6 Private Automobile Attributes

- We illustrate but a few attributes:

245. Automobiles have static number plate registration numbers.
246. Automobiles have dynamic positions on the road net:

    [243a] either *at a hub* identified by some h_ui,

    [243b] or *on a link*, some *fraction, frac:Fract* down an *identified link, l_ui*, from one of its *identified connecting hub*s, fh_ui, in the direction of the other *identified hub*, th_ui.

type

```
245  RegNo
246  APos    == atHub | onLink
243a atHub   :: h_ui:H_UI
243b onLink  :: fh_ui:H_UI × l_ui:L_UI × frac:Fract × th_ui:H_UI
243b Fract   = Real, axiom frac:Fract · 0<frac<1
```

value

```
245  attr_RegNo: A → RegNo
246  attr_APos: A → APos
```

- Obvious attributes that are not illustrated are those of
  - velocity and acceleration,
  - forward or backward movement,
  - turning right, left or going straight,
  - etc.

- The *acceleration, deceleration, even velocity,* or *turning right, turning left, moving straight*, or *forward* or *backward* are seen as *command actions*.
  - As such they denote actions by the automobile —
  - such as pressing the accelerator, or lifting accelerator pressure or *braking*, or *turning the wheel* in one direction or another, etc.
  - As actions they have a kind of counterpart in the velocity, the acceleration, etc. attributes.

- Observe that bus companies each have their own distinct *bus time table*, and that these are modeled as *programmable*, Item 240 on Slide 490, page 490.
- Observe then that buses each have their own distinct *bus time table*, and that these are model-led as *inert*, Item 242 on Slide 491, page 491.

- In Items Sli. 260 and Sli. 264, we illustrated an aspect of domain analysis & description that may seem, and at least some decades ago would have seemed, strange: namely that if we can think, hence speak, about it, then we can model it “as a fact” in the domain. The case in point is that we include among hub and link attributes their histories of the timed whereabouts of buses and automobiles.[7]

[7] In this day and age of road cameras and satellite surveillance these traffic recordings may not appear so strange: We now know, at least in principle, of technologies that can record approximations to the hub and link traffic attributes.

A.3.3.7 Intentionality

247. Seen from the point of view of an automobile there is its own traffic history, A_Hist, which is a (time ordered) sequence of timed automobile's positions;
248. seen from the point of view of a hub there is its own traffic history, H_Traffic Item Sli. 260, which is a (time ordered) sequence of timed maps from automobile identities into automobile positions; and
249. seen from the point of view of a link there is its own traffic history, L_Traffic Item Sli. 264, which is a (time ordered) sequence of timed maps from automobile identities into automobile positions.

- The *intentional "pull"* of these manifestations is this:

250. The union, i.e. proper merge of all automobile traffic histories, AllATH, must now be identical to the same proper merge of all hub, AllHTH, and all link traffic histories, AllLTH.

```
type
247   A_Hi  = (𝕋 × APos)*
234   H_Trf = A_UI →m (𝕋𝕀𝕄𝔼 × APos)*
238   L_Trf = A_UI →m (𝕋𝕀𝕄𝔼×APos)*
250   AllATH=𝕋𝕀𝕄𝔼 →m(AUI →mAPos)
250   AllHTH=𝕋𝕀𝕄𝔼 →m(AUI →mAPos)
250   AllLTH =𝕋𝕀𝕄𝔼 →m(AUI →mAPos)
axiom
250   let allA=mrg_AllATH({(a,attr_A_Hi(a))|a:A·a ∈ as}),
250      allH=mrg_AllHTH({attr_H_Trf(h)|h:H·h ∈ hs}),
250      allL =mrg_AllLTH({attr_L_Trf(l)|l:L·h ∈ ls}) in
250   allA = mrg_HLT(allH,allL) end
```

- We leave the definition of the merge functions to the listener !
  - We endow
    - ⋆ each automobile with its history of timed positions and
    - ⋆ each hub and link with their histories of timed automobile positions.
  - These histories are facts !
  - They are not something that is laboriously recorded, where such recordings may be imprecise or cumbersome[8].
  - The facts are there, so we can (but may not necessarily) talk about these histories as facts.

[8] or thought technologically in-feasible – at least some decades ago!

- It is in that sense that the purpose (`transport′)
  - for which man let automobiles, hubs and link be made
  - with their `transport′ intent
  - are subject to an *intentional "pull"*.

- *It can be no other way: if automobiles "record" their history, then hubs and links must together "record" identically the same history!*.

Intentional Pull – General Transport:

- These are examples of human intents:
  - they create *roads* and *automobiles* with the intent of *transport*,
  - they create *houses* with the intents of *living, offices, production*, etc., and
  - they create *pipelines* with the intent of *oil* or *gas transport* ■

A.4 Perdurants

- In this section we transcendentally "morph" **parts** into **behaviours.**
- We analyse that notion and its constituent notions of
  - **actors**,
  - **channels** and **communication**,
  - **actions** and
  - **events**.
- The main transcendental deduction of this chapter
  - is that of associating
  - with each part
  - a behaviour.
- This section shows the details of that association.

- Perdurants are understood in terms of
  - a notion of ***state*** and
  - a notion of ***time***.

State Values versus State Variables:

- Item 197 on Slide 466 expresses the **value** of all parts of a road transport system:

197. ps:(UoB|H|L|BC|B|A)-**set** $\equiv rts \cup hls \cup bcs \cup bs \cup as$.

251. We now introduce the set of variables, one for each part value of the domain being modeled.

251. { **variable** vp:(UoB|H|L|BC|B|A) | vp:(UoB|H|L|BC|B|A) $\cdot$ $vp \in ps$ }

Buses and Bus Companies

- A bus company is like a "root" for its fleet of "sibling" buses.
- But a bus company may cease to exist without the buses therefore necessarily also ceasing to exist.
- They may continue to operate, probably illegally, without, possibly.
  a valid bus driving certificate.
- Or they may be passed on to either private owners or to other bus companies.
- We use this example as a reason for not endowing a "block structure" concept on behaviours.

A.4.1 Channels and Communication

A.4.1.1 Channel Message Types

- We ascribe types to the messages offered on channels.

252. Hubs and links communicate, both ways, with one another, over channels, `hl_ch`, whose indexes are determined by their mereologies.
253. Hubs send one kind of messages, links another.
254. Bus companies offer timed bus time tables to buses, one way.
255. Buses and automobiles offer their current, timed positions to the road element, hub or link they are on, one way.

type
253 H_L_Msg, L_H_Msg
252 HL_Msg = H_L_Msg | L_F_Msg
254 BC_B_Msg = T × BusTimTbl
255 V_R_Msg = T × (BPos|APos)

A.4.1.2 Channel Declarations

256. This justifies the channel declaration which is calculated to be:

channel
256 { hl_ch[h_ui,l_ui]:H_L_Msg | h_ui:H_UI,l_ui:L_UI·i ∈ $h_{ui}s$∧j ∈ $lh_{ui}m$(h_ui) }
256 ∪
256 { hl_ch[h_ui,l_ui]:L_H_Msg | h_ui:H_UI,l_ui:L_UI·l_ui ∈ $l_{ui}s$∧i ∈ $lh_{ui}m$(l_ui) }

- We shall argue for bus company-to-bus channels based on the mereologies of those parts.
  - Bus companies need communicate to all its buses, but not the buses of other bus companies.
  - Buses of a bus company need communicate to their bus company, but not to other bus companies.

257. This justifies the channel declaration which is calculated to be:

channel
257 { bc_b_ch[bc_ui,b_ui] | bc_ui:BC_UI, b_ui:B_UI · bc_ui ∈ $bc_{ui}s$ ∧ b_ui ∈ $b_{ui}s$ }: BC_B_Msg

- We shall argue for vehicle to road element channels based on the mereologies of those parts.
  - Buses and automobiles need communicate to
    - all hubs and
    - all links.

258. This justifies the channel declaration which is calculated to be:

channel
258 { v_r_ch[v_ui,r_ui] | v_ui:V_UI,r_ui:R_UI · v_ui∈ $v_{ui}s$∧r_ui∈ $r_{ui}s$ }: V_R_Msg

A.4.2 Behaviours

A.4.2.1 Road Transport Behaviour Signatures

- We first decide on names of behaviours.
  - In the translation schemas
  - we gave schematic names to behaviours
  - of the form $\mathcal{M}_P$.
- We now assign mnemonic names:
  - from part names to names of transcendentally interpreted behaviours
  - and then we assign signatures to these behaviours.

A.4.2.1.1 Hub Behaviour Signature

259. $\mathsf{hub}_{h_{ui}}$:
    (a) there is the usual “triplet” of arguments: unique identifier, mereology and static attributes;
    (b) then there are the programmable attributes;
    (c) and finally there are the input/output channel references: first those allowing communication between hub and link behaviours,
    (d) and then those allowing communication between hub and vehicle (bus and automobile) behaviours.

value

259 $\text{hub}_{h_{ui}}$:

259a h_ui:H_UI×(vuis,luis,_):H_Mer×HΩ

259b → (HΣ×H_Traffic)

259c → **in**,**out** { h_l_ch[h_ui,l_ui] | l_ui:L_UI·l_ui ∈ luis }

259d { ba_r_ch[h_ui,v_ui] | v_ui:V_UI·v_ui∈vuis } **Unit**

259a **pre**: vuis = $v_{ui}s$ ∧ luis = $l_{ui}s$

A.4.2.1.2 Link Behaviour Signature

260. $\text{link}_{l_{ui}}$:
   (a) there is the usual “triplet” of arguments: unique identifier, mereology and static attributes;
   (b) then there are the programmable attributes;
   (c) and finally there are the input/output channel references: first those allowing communication between hub and link behaviours,
   (d) and then those allowing communication between link and vehicle (bus and automobile) behaviours.

value

260 link$_{l_{ui}}$:
260a l_ui:L_UI×(vuis,huis,_):L_Mer×LΩ
260b → (LΣ×L_Traffic)
260c → **in**,**out** { h_l_ch[h_ui,l_ui] | h_ui:H_UI:h_ui ∈ huis }
260d { ba_r_ch[l_ui,v_ui] | v_ui:(B_UI|A_UI)·v_ui∈vuis } **Unit**
260a **pre**: vuis = $v_{ui}s$ ∧ huis = $h_{ui}s$

A.4.2.1.3 Bus Company Behaviour Signature

261. $\text{bus\_company}_{bc_{ui}}$:
    (a) there is here just a "doublet" of arguments: unique identifier and mereology;
    (b) then there is the one programmable attribute;
    (c) and finally there are the input/output channel references allowing communication between the bus company and buses.

value

261 bus_company$_{bc_{ui}}$:
261a bc_ui:BC_UI×(_,_,buis):BC_Mer
261b → BusTimTbl
261c **in,out** {bc_b_ch[bc_ui,b_ui]|b_ui:B_UI·b_ui∈buis} **Unit**
261a **pre**: buis = $b_{ui}s$ ∧ huis = $h_{ui}s$

A.4.2.1.4 Bus Behaviour Signature

262. $\mathsf{bus}_{b_{ui}}$:
    (a) there is here just a “doublet” of arguments: unique identifier and mereology;
    (b) then there are the programmable attributes;
    (c) and finally there are the input/output channel references: first the input/output allowing communication between the bus company and buses,
    (d) and the input/output allowing communication between the bus and the hub and link behaviours.

value

```
262  bus_{b_ui}:
262a  b_ui:B_UI×(bc_ui,_,ruis):B_Mer
262b    → (LN × BTT × BPOS)
262c    → out bc_b_ch[ bc_ui,b_ui ],
262d        {ba_r_ch[ r_ui,b_ui ]|r_ui:(H_UI|L_UI)·ui∈v_ui s} Unit
262a    pre: ruis = r_ui s ∧ bc_ui ∈ bc_ui s
```

A.4.2.1.5 Automobile Behaviour Signature

263. automobile$_{a_{ui}}$:
   (a) there is the usual "triplet" of arguments: unique identifier, mereology and static attributes;
   (b) then there is the one programmable attribute;
   (c) and finally there are the input/output channel references allowing communication between the automobile and the hub and link behaviours.

value

263 automobile$_{a_{ui}}$:

263a a_ui:A_UI×(_,_,ruis):A_Mer×rn:RegNo

263b → apos:APos

263c **in**,**out** {ba_r_ch[a_ui,r_ui]|r_ui:(H_UI|L_UI)·r_ui∈ruis} **Unit**

263a **pre**: ruis = $r_{ui}s$ ∧ a_ui ∈ $a_{ui}s$ ■

A.4.2.2 Behaviour Definitions

- We only illustrate automobile, hub and link behaviours.

A.4.2.2.1 Automobile Behaviour at a Hub

- We define the behaviours in a different order than the treatment of their signatures.
- We "split" definition of the automobile behaviour
  - into the behaviour of automobiles when positioned at a hub, and
  - into the behaviour automobiles when positioned at on a link.
  - In both cases the behaviours include the "idling" of the automobile, i.e., its "not moving", standing still.

264. We abstract automobile behaviour at a Hub (hui).
265. The vehicle remains at that hub, “idling”,
266. informing the hub behaviour,
267. or, internally non-deterministically,
    (a) moves onto a link, tli, whose “next” hub, identified by th_ui, is obtained from the mereology of the link identified by tl_ui;
    (b) informs the hub it is leaving and the link it is entering of its initial link position,
    (c) whereupon the vehicle resumes the vehicle behaviour positioned at the very beginning (0) of that link,
268. or, again internally non-deterministically,
269. the vehicle “disappears — off the radar” !

```
264  automobile_a_ui(a_ui,({},(ruis,vuis),{}),rn)
264          (apos:atH(fl_ui,h_ui,tl_ui)) ≡
265      (ba_r_ch[a_ui,h_ui] ! (record_TIME(),atH(fl_ui,h_ui,tl_ui));
266       automobile_a_ui(a_ui,({},(ruis,vuis),{}),rn)(apos))
267     ⊓
267a      (let ({fh_ui,th_ui},ruis′)=mereo_L(℘(tl_ui)) in
267a           assert: fh_ui=h_ui ∧ ruis=ruis′
264       let onl = (tl_ui,h_ui,0,th_ui) in
267b      (ba_r_ch[a_ui,h_ui] ! (record_TIME(),onL(onl)) ||
267b       ba_r_ch[a_ui,tl_ui] ! (record_TIME(),onL(onl))) ;
267c       automobile_a_ui(a_ui,({},(ruis,vuis),{}),rn)
267c            (onL(onl)) end end)
268     ⊓
269      stop
```

A.4.2.2.2 Automobile Behaviour On a Link

270. We abstract automobile behaviour on a Link.
    (a) Internally non-deterministically, either
        i. the automobile remains, “idling”, i.e., not moving, on the link,
        ii. however, first informing the link of its position,
    (b) or
        i. **if** if the automobile’s position on the link *has not yet reached the hub*, **then**
            A. then the automobile moves an arbitrary small, positive **Real**-valued *increment* along the link
            B. informing the hub of this,
            C. while resuming being an automobile ate the new position, or

ii. **else**,
   A. while obtaining a "next link" from the mereology of the hub (where that next link could very well be the same as the link the vehicle is about to leave),
   B. the vehicle informs both the link and the imminent hub that it is now at that hub, identified by th_ui,
   C. whereupon the vehicle resumes the vehicle behaviour positioned at that hub;

(c) or

(d) the vehicle "disappears — off the radar" !

```
270        automobile_{a_ui}(a_ui,({},ruis,{}),rno)
270                      (vp:onL(fh_ui,l_ui,f,th_ui)) ≡
270(a)ii     (ba_r_ch[ thui,aui ]!atH(lui,thui,nxt_lui) ;
270(a)i       automobile_{a_ui}(a_ui,({},ruis,{}),rno)(vp))
270b        ⊓
270(b)i     (if not_yet_at_hub(f)
270(b)i        then
270(b)iA          (let incr = increment(f) in
264               let onl = (tl_ui,h_ui,incr,th_ui) in
270(b)iB            ba−r_ch[ l_ui,a_ui ] ! onL(onl) ;
270(b)iC            automobile_{a_ui}(a_ui,({},ruis,{}),rno)
270(b)iC                              (onL(onl))
270(b)i           end end)
270(b)ii        else
270(b)iiA         (let nxt_lui:L_UI•nxt_lui ∈ mereo_H(℘(th_ui)) in
270(b)iiB          ba_r_ch[ thui,aui ]!atH(l_ui,th_ui,nxt_lui) ;
270(b)iiC          automobile_{a_ui}(a_ui,({},ruis,{}),rno)
270(b)iiC                             (atH(l_ui,th_ui,nxt_lui)) end)
270(b)i       end)
270c        ⊓
270d          stop
270(b)iA   increment: Fract → Fract
```

A.4.2.2.3 Hub Behaviour

271. The hub behaviour
    (a) non-deterministically, externally offers
    (b) to accept timed vehicle positions —
    (c) which will be at the hub, from some vehicle, v_ui.
    (d) The timed vehicle hub position is appended to the front of that vehicle's entry in the hub's traffic table;
    (e) whereupon the hub proceeds as a hub behaviour with the updated hub traffic table.
    (f) The hub behaviour offers to accept from any vehicle.
    (g) A **post** condition expresses what is really a **proof obligation**: that the hub traffic, ht′ satisfies the **axiom** of the endurant hub traffic attribute Item Sli. 260.

```
value
271  hub_{h_ui}(h_ui,(,(luis,vuis)),hω)(hσ,ht) ≡
271a    ⌷
271b      { let m = ba_r_ch[ h_ui,v_ui ] ? in
271c          assert: m=(_,atHub(_,h_ui,_))
271d        let ht′ = ht † [ h_ui ↦ ⟨m⟩⁀ht(h_ui) ] in
271e        hub_{h_ui}(h_ui,(,(luis,vuis)),(hω))(hσ,ht′)
271f      | v_ui:V_UI•v_ui∈vuis end end }
271g   post: ∀ v_ui:V_UI•v_ui ∈ dom ht′⇒time_ordered(ht′(v_ui))
```

A.4.2.2.4 Link Behaviour

272. The link behaviour non-deterministically, externally offers
273. to accept timed vehicle positions —
274. which will be on the link, from some vehicle, v_ui.
275. The timed vehicle link position is appended to the front of that vehicle's entry in the link's traffic table;
276. whereupon the link proceeds as a link behaviour with the updated link traffic table.
277. The link behaviour offers to accept from any vehicle.
278. A **post** condition expresses what is really a **proof obligation**: that the link traffic, lt′ satisfies the **axiom** of the endurant link traffic attribute Item Sli. 264.

```
272 link_{l_ui}(l_ui,(_,(huis,vuis),_),lω)(lσ,lt) ≡
272   □
273     { let m = ba_r_ch[l_ui,v_ui] ? in
274           assert: m=(_,onLink(_,l_ui,_,_))
275       let lt′ = lt † [l_ui ↦ ⟨m⟩⌢lt(l_ui)] in
276       link_{l_ui}(l_ui,(huis,vuis),hω)(hσ,lt′)
277     | v_ui:V_UI•v_ui∈vuis end end }
278   post: ∀ v_ui:V_UI•v_ui ∈ dom lt′⇒time_ordered(lt′(v_ui))
```

A.5 System Initialisation

A.5.1 Initial States

value

hs:H-**set** ≡ ≡ obs_sH(obs_SH(obs_RN(rts)))
ls:L-**set** ≡ ≡ obs_sL(obs_SL(obs_RN(rts)))
bcs:BC-**set** ≡ obs_BCs(obs_SBC(obs_FV(obs_RN(rts))))
bs:B-**set** ≡ ∪{obs_Bs(bc)|bc:BC·bc ∈ bcs}
as:A-**set** ≡ obs_BCs(obs_SBC(obs_FV(obs_RN(rts))))

A.5.2 Initialisation

- We are reaching the end of this domain modeling example.
  - Behind us there are narratives and formalisations.
  - Based on these we now express
    - the signature and
    - the body of the definition
  - of a *"system build and execute"* function.

279. The system to be initialised is
    (a) the parallel compositions (‖) of
    (b) the distributed parallel composition (‖{...|...}) of all hub behaviours,
    (c) the distributed parallel composition (‖{...|...}) of all link behaviours,
    (d) the distributed parallel composition (‖{...|...}) of all bus company behaviours,
    (e) the distributed parallel composition (‖{...|...}) of all bus behaviours, and
    (f) the distributed parallel composition (‖{...|...}) of all automobile behaviours.

value

```
279  initial_system: Unit → Unit
279  initial_system() ≡
279b    ∥ { hub_{h_ui}(h_ui,me,hω)(htrf,hσ)
279b      | h:H•h ∈ hs, h_ui:H_UI•h_ui=uid_H(h), me:HMetL•me=mereo_H(h),
279b        htrf:H_Traffic•htrf=attr_H_Traffic_H(h),
279b        hω:HΩ•hω=attr_HΩ(h), hσ:HΣ•hσ=attr_HΣ(h)∧hσ ∈ hω }
279a    ∥
279c    ∥ { link_{l_ui}(l_ui,me,lω)(ltrf,lσ)
279c        l:L•l ∈ ls, l_ui:L_UI•l_ui=uid_L(l), me:LMet•me=mereo_L(l),
279c        ltrf:L_Traffic•ltrf=attr_L_Traffic_H(l),
279c        lω:LΩ•lω=attr_LΩ(l), lσ:LΣ•lσ=attr_LΣ(l)∧lσ ∈ lω }
279a    ∥
279d    ∥ { bus_company_{bc_ui}(bcui,me)(btt)
279d        bc:BC•bc ∈ bcs, bc_ui:BC_UI•bc_ui=uid_BC(bc), me:BCMet•me=mereo_BC(bc),
279d        btt:BusTimTbl•btt=attr_BusTimTbl(bc) }
279a    ∥
279e    ∥ { bus_{b_ui}(b_ui,me)(ln,btt,bpos)
279e        b:B•b ∈ bs, b_ui:B_UI•b_ui=uid_B(b), me:BMet•me=mereo_B(b), ln:LN:pln=attr_LN(b),
279e        btt:BusTimTbl•btt=attr_BusTimTbl(b), bpos:BPos•bpos=attr_BPos(b) }
279a    ∥
279f    ∥ { automobile_{a_ui}(a_ui,me,rn)(apos)
279f        a:A•a ∈ as, a_ui:A_UI•a_ui=uid_A(a), me:AMet•me=mereo_A(a),
279f        rn:RegNo•rno=attr_RegNo(a), apos:APos•apos=attr_APos(a) }  ■
```